A comparison of learner and designer models in the use of direct manipulation educational software in the context of learning about interacting variables in photosynthesis.

Squires, David John

A Comparison of Learner and Designer Models in the Use of Direct Manipulation Educational Software in the Context of Learning About Interacting Variables in Photosynthesis

David John Squires

Thesis submitted in fulfilment of the requirements for the degree of PhD of the University of London

King's College London, Centre for Educational Studies

May 1994

Abstract

The work involved an in-depth evaluation of the design and use of a sophisticated educational software package in an area known to be conceptually difficult. The software (*Bioview*) allows the exploration of the relationship between three interacting variables by the direct manipulation of a pictorial representation of these variables. Limiting factors in photosynthesis was chosen as a exemplar task domain. An extensive model (the Jigsaw Model) for the assessment of cognitive aspects of educational software was developed as a basis for the evaluation.

The focus of the research was on the use of *Bioview* by three pairs of 6th form students to answer set questions about limiting factors in photosynthesis. In each pair a student was asked to teach the other student to use *Bioview* to answer the set questions. These teaching sessions were video-taped and the audio component of each video recording was transcribed. In addition audio recordings were made of (i) a teaching session in which the "teacher-students" were introduced to *Bioview*, and (ii) group interviews with the teacher-students and the paired students. Other data sources included package documentation and reports of preliminary studies conducted with Masters students.

The video records form the prime data source. The "Goals, Operators, Methods, and Selection" (GOMS) model was used as an initial basis to analyse the human-computer interaction in these sessions. The Jigsaw Model was used to complete a critique of the design of *Bioview*. This critique has been used as a basis for a consideration of learner and designer models in the use of direct manipulation educational software, leading to the identification of four design paradoxes in this context; a black box paradox, a control paradox, a display paradox, and an interaction paradox; all of which need to be considered in the design of direct manipulation educational software.

Acknowledgements

I would like to thank my supervisor, Professor David Johnson, for his continual support and encouragement. He was always available to give advice, and his perceptive comments and criticism were invaluable.

Thanks are due to Delia, Sharon, Ruth, Alice, Uri, and Tom for being research subjects, and to Eugene McConlough and Barbara Thompson, both staff members of Wakelys School, for helping me identify such willing research subjects. The Masters students who followed the Design and Evaluation of Educational Software module that I taught in 1991 provided me with some very perceptive insights which greatly assisted the planning of my research programme. My thanks go to them for their indulgence in conducting a preliminary evaluation of *Bioview* for me. In particular I would like to thank Barbara Thompson and Tony Hill for allowing me to use their reports on the evaluation of *Bioview* as data.

Dr Anne McDougall kindly read a draft of the thesis during the final stages of its preparation, and Professor Arthur Lucas read early versions of Chapter 2 and 3. They both made very helpful comments for which I am very grateful. In retrospect thanks are also due to the other members of the *Bioview* design team (Colin Smith, Dr Royston Sellman, Sophie McCormick, and David Riley) for the stimulating and formative discussions we had when we were developing this package. I am also grateful to Alan Edis for checking the coding of the video transcript data.

Finally, thanks to my family for all their support and encouragement. Emer and Orla, despite their tender years, have been very understanding of a "distant" father, and Tracy has been outstanding in looking after them during this period. My parents, as always, have helped and supported me. Most importantly, my wife, Deirdre, has provided support and encouragement, without which I would never have had the resolve to complete this thesis.

Contents

Figures

Tables

Chapter 1

Introduction

This thesis is concerned with a comparison of designer and learner models of educational software, with particular respect to the use of software based on the direct manipulation interface paradigm. *Bioview* (Squires, 1991), a direct manipulation educational software application specifically designed for use in education, provides a software context for this comparison. Interacting variables in photosynthesis, an area which is acknowledged to be conceptually difficult, provides a task context. The comparison of learner and designer models is used to develop a comprehensive framework for the evaluation of direct manipulation educational software.

The notion of using mental models to represent human-computer interaction is relatively new, and there is still debate about what aspects of human-computer interaction should be modelled, what form the model should take, and what purpose the model should fulfil. The scope of the application of human-computer interaction models raises the question of whom the model purports to represent; most significantly, the designer or the learner. Allied to this is the question of what aspects of the software environment the model represents - the task itself, the interface to the user presented by the system, or the basic system architecture? Models of human-computer interaction can take various forms, for example a surrogate representation (Young, 1981), a metaphor (Carroll & Thomas, 1982), a "glass box machine" (duBoulay, O'Shea & Monk, 1981), or a combination of approaches such as general transition networks (Kieras & Polson, 1985). Each type of model has its strengths and weaknesses and the choice of a model will depend on the purpose envisaged for its use. This purpose can be broadly described as descriptive or prescriptive (Carroll & Olson, 1988). Descriptive models attempt to represent what users actually know, and as such their primary function is to provide a framework to explain the observed use of software. Prescriptive or competence models are concerned with what users should know, and they are used to provide representations of how the designer intends software to be used.

The eclectic nature of human-computer interaction models makes it difficult to clarify a basic framework for the evaluation of software. A good evaluation model should incorporate models which take account of the learner and the designer, the task and the system, and be both descriptive and prescriptive. A comparison of learner and designer models is used in this thesis to put together the different pieces of such a comprehensive framework to propose a "Jigsaw Model" for the evaluation of direct manipulation educational software.

1.1 The Jigsaw Model for the evaluation of educational software

The models that have been advocated in the literature for human-computer interaction are broadly reviewed in Chapter 2. This is a wide ranging area of development in human-computer interaction studies, and this review does not claim to be exhaustive. However, it is intended that the review will indicate the major forms that human-computer interaction models can take.

Many existing models were developed on the basis of research into the use of command driven interfaces in which the user input and program output are separately conceived. Direct manipulation has now replaced command execution as the dominant form of human-computer interaction in state of the art interfaces. As an essential feature of direct manipulation is the use of the same object as both an input and output device, these models having limited application to the use of direct manipulation software. The defining aspects of the direct manipulation paradigm are considered in Chapter 2 in terms of the user-system interaction cycles described by Norman (1986) and Abowd & Beale (1991) as a prelude to identifying the specific weaknesses of existing models in a direct manipulation context. This analysis is used to identify an existing model as the basis for the development of a comprehensive evaluation framework as described above.

The model chosen as a basis for expansion is the Goals, Operators, Methods and Selection model (GOMS) developed by Card, Moran & Newell (1983). This is a well established competence model, with many reports in the research literature of its use and modification. The major extension of the GOMS model required for the development of a comprehensive evaluation framework is the linking of a compatible descriptive model. The development of such a model and its linking to the GOMS model to form the Jigsaw model is described in Chapter 2.

1.2 A context for applying the Jigsaw Model

The non-routine cognitive nature of educational tasks indicates that models for the evaluation of educational software should take account of exploratory open-ended cognitive activity. However, most of the research in human-computer interaction that has considered user models has been concerned with the routine use of well known systems such as word-processors and spreadsheets to complete well defined tasks. The cognition involved in the completion of these routine tasks is based on the use of internalised and well understood procedures, for example moving a block of text in a document. Experienced users will complete these tasks in well ordered concise ways without the need for exploratory problem solving use of the system. This contrasts with the use of educational software when the user typically has an incomplete or misconceived understanding of the task in question. Educational software is designed to assist students in the development of understanding and the correction of misconceptions - a non-routine cognitive task.

A non-routine cognitive dimension to the evaluation has been introduced by including a consideration of the typical misconceptions reported in the research literature in understanding how the rate of photosynthesis is limited by interacting variables. Specifically, the research subjects were asked to explore the relationships between the principal variables which interact to determine the rate of photosynthesis of a green plant. Research into students' understanding of interacting variables in general, and how environmental factors interact to limit the rate of photosynthesis in particular, is well documented as described in Chapter 3. The research literature shows that most students have a very limited appreciation of the notion of interacting variables. Typically they use a "change one variable at a time while all other variables are fixed" paradigm. While this may be appropriate in the physical sciences, it is very often inappropriate in biology when the *interaction* between variables is often critical. The way in which the light intensity, level of carbon dioxide and atmospheric temperature interact to limit the rate of photosynthesis provides a classic example of the importance of considering the interaction between variables.

Bioview provides a way of exploring the relationships between three interacting variables through the direct manipulation of a pictorial representation of the variables, as described in Chapter 4. The results of this direct manipulation can be represented by instances of graphs of the rate of photosynthesis plotted against each of these variables. It is possible to display multiple instances and animated sequences of these graphs, enabling comparisons of the rate for different sets of values of the variables. The software runs in Microsoft *Windows*, enabling the display of multiple windows on the screen and links to other software applications.

1.3 Observation of the use of *Bioview*

As there is very little reported research concerned with human-computer interaction and the use of educational software, it follows that focal research issues in this area are not evident, and that there are no established research methodologies which are obviously applicable in this context. Therefore, it was impossible to make a priori decisions about appropriate research questions and an appropriate research methodology. Consequently, a preliminary phase of research was completed in order to clarify an initial research agenda and a broadly defined methodology (see Chapter 6). The initial research agenda and methodology were refined as new research questions and data analysis techniques were suggested by the results of data analysis. The finally established methodology is described in Chapter 5.

The preliminary phase of research was centred on a questionnaire based heuristic evaluation of *Bioview* by a group of experienced teachers. The questionnaire items only suggested areas of concern and the teachers were expected to evaluate *Bioview* simply on the basis of their general experience of using educational software (see Appendix 4). After the evaluation the group met to discuss the results of their evaluations. Analysis of the questionnaire returns and the transcript of this group discussion indicated that the successful use of direct manipulation techniques in a *Windows* environment was considered to be problematic for inexperienced users. In particular, a number of comments were made about the potential problems associated with the interpretation and use of the "datacube" object as the focus for direct manipulation.

Two of the participating teachers also completed small empirical studies based on their practical use of *Bioview*. These reports were analysed with particular reference to the issues raised by an analysis of the questionnaire returns and the group discussion transcript. This analysis confirmed the perception that direct manipulation in a *Windows* environment should be a prominent feature of the research agenda. One of these reports described the problems encountered in recording the observations of the use of *Bioview*. The teacher reported that it was difficult to record the operations that the students executed and it was difficult to link these operations to the student's intentions. The identification of these problems was instrumental in establishing a data collection technique for the second phase of the research. It was decided to base the data collection on making a video record of the observed use of *Bioview*. This ensured that a faithful and complete record of the human-computer interaction was made. In order to prompt the articulation of user intentions, observations were made of the use of *Bioview* by pairs of students, with one of the students given the responsibility of teaching the other student to use *Bioview* to answer a number of set questions. It was anticipated that this would encourage the teacher-students to articulate their understanding of the use of *Bioview* and the other students to express their conceptual problems.

In preparation, for the conduct of the observation programme planned for the second phase of the research, relevant sections of the Users' and Teachers Guides for *Bioview* were inspected. In particular a definition of the task domain was determined by an analysis of the set questions, which consisted of an adapted version of the set of questions appearing in the Teachers' Guide. This analysis allowed "successful" methods corresponding to the designer's model to be identified.

During the second phase, the use of *Bioview* by three pairs of 6th form students to answer five set questions about limiting factors in photosynthesis was observed. As explained above, in each pair a student was asked to teach the other student to use *Bioview* to answer the set questions. These teaching sessions were video-taped as planned and the audio component of each video recording was transcribed. Group interviews with the "teacher-students" and the paired students were conducted after the observed sessions had all been completed.

Data analysis techniques for the data collected during this phase were developed for an analysis in terms of the designer's and learner's models from a task perspective and a system perspective. An analysis of the application of the GOMS model was also completed. An inspection of the laboratory session descriptions (Appendix 6) suggested appropriate forms of data reduction from both a task and system perspective. As the data analysis was completed the analysis techniques were refined to take account of emerging results. The data analysis completed in the second phase is described in Chapter 7.

The analysis from a system perspective resulted in three findings. Firstly, the state of the system, as represented by the state of the display, was very influential in the selection of methods. Secondly, the users needed informative confirmation of the effects of their actions for a smooth flow of activity between the task and software domains. Thirdly, users misunderstood the relationship between the current state of the system and the representation of this state by the display. This last finding contrasts with the superficial impression of a close match between learner and designer models implied by the users' extensive selection of successful methods. The analysis from a task perspective provided little evidence that the *interaction* between variables was considered by the users. In particular it appeared the learners did not possess, or were incapable of applying, an understanding of the concept of limiting factors in photosynthesis. There was some evidence of preferential application of direct manipulation techniques with respect to the use of this technique for fixing the value of a variable, or exploring the effect of varying the value of a variable. The application of the GOMS approach to describe the human-computer interaction in the laboratory sessions was effective as a method for describing the interaction at a superficial level.

1.4 A comparison of learner and designer models

In Chapter 8 the use of the Jigsaw Model to identify and explain likely design issues concerned with the use of *Bioview* to explore limiting factors in photosynthesis is discussed. The model is applied to specifically address the findings reported in Chapter 7 to complete a critique of *Bioview*, which is used to make some specific design recommendations. In addition, the critique is used to comment generally on the comparison between learner and designer models that may be employed during the use of direct manipulation educational software, leading to the identification of four design paradoxes which need to be addressed in this design context.

Chapter 2

Evaluation of direct manipulation educational software

Many attempts have been made to devise descriptive frameworks which can be used as a basis for the evaluation of educational software. These frameworks can be classified as (i) classification systems based on categories to which educational software applications might be assigned, (ii) schemes which emphasise the roles that software is intended to play, and (iii) schemes which relate software to commonly accepted educational rationales (Squires & McDougall, 1994). A consideration of each type of framework shows that none of them attempt to consider the significance of the nature of human-computer interaction in the learning process; an area of increasing importance, given recent developments in interface design, such as the advent of direct manipulation interfaces.

The literature provides examples of frameworks based on classification by application type (see for example Beech, 1983; Hofmeister, 1984; Salvas & Thomas, 1984; Wellington, 1985; Newman, 1988; Bitter & Camuse, 1988; USA Office of Technology Assessment, 1988; Organisation for Economic Co-operation and Development, 1989; Simpson & Thompson, 1990; Pelgrum & Plomp 1991). For example, Pelgrum & Plomp include the following in their list: drill and practice, tutorial programs, word processing, painting and drawing, music composition, simulation, recreational games, educational games, programming languages, spreadsheet, mathematics graphing, statistics, database, lab interfaces, programs to control devices, programs to control interactive video, CAD/CAM, CAI authoring language, item banks, record/score tests, grade book, computer communication, and tools/utilities. It is clear that this approach is simply classificatory with little or no scope for including a consideration of human-computer interaction.

A classic example of classification by educational role is provided by Taylor (1980), who described three roles for educational software: tutor, tool, and tutee. According to Taylor software operates as a tutor when the "computer presents some subject material, the student responds, the computer evaluates the response, and, from the results of the evaluation, determines what to present next" (Taylor, 1980, p.3). Drill and practice exercises and adaptive tutorial programs are typical of this software role. The tool mode is characterised by the computer's performance of tedious labour intensive activities, enabling the learner to concentrate on essential concepts without being distracted by the demands of "inauthentic" labour. As Taylor remarks, "to function as a tool, the classroom computer need only have some useful capability programmed into it such as statistical analysis, super calculation, or word processing" (Taylor, 1980, p. 4). The tutee mode refers to provision by the computer of environments in which learners can "teach" the computer through expressing their own ideas and solutions to problems. O'Shea & Self (1983) also describe a framework based on classification by educational role. They classify software as either acting as a surrogate teacher or as a learning resource. Software acting as a surrogate teacher prescribes tasks and presents a body of knowledge. Software acting as a resource places a far greater emphasis on learning by involvement. Self (1985) develops this framework further by considering educational software in terms of the analysis by Rowntree (1982) of the functions of educational media. It is clear from these examples that human-computer interaction does not feature as an evaluation issue.

Kemmis, Atkin, & Wright (1977) and Chandler (1984) provide examples of frameworks based on classification by educational rationale. Despite its age the framework proposed by Kemmis et al. is still popular. It is based on the proposition of three paradigms of education "through which we may grasp the major ways in which the developers of computer assisted learning conceive the curriculum task" (Kemmis et al., 1977, p. 24). These paradigms are labelled as (i) instructional, which aims explicitly to teach material, usually by breaking it up into small parts and presenting these to students; (ii) revelatory, which emphasises learning by discovery, with software used to provide environments for exploration and discovery, and (iii) conjectural, which includes the articulation and manipulation of ideas and hypothesis testing. Kemmis et al. suggest a fourth paradigm - the emancipatory paradigm - which originates from the notion of the computer as a labour saving tool, and which only exists in conjunction with one of the other three. The example provided by Kemmis et al. indicates that these frameworks are primarily concerned with the features of the educational task in question, not the details of the associated human-computer interaction.

It is clear that existing frameworks do not address human-computer interaction issues. This needs to be rectified. Modern user interfaces are commonly used in schools, and a credible evaluation model for the cognition associated with the use of educational software needs to take due account of both educational and interaction design issues.

In this chapter the evaluation of direct manipulation, possibly the most popular current design paradigm, is considered as a prelude to research into its use in an educational context. Section 2.1 describes a general framework for human-computer interaction, which is applied in Section 2.2 to the particular case of direct manipulation. Section 2.3 reviews common models of human-computer interaction. In Section 2.4 the problems associated with applying existing models to human computer interaction are considered. An existing model is identified to represent direct manipulation, prior to the development in Section 2.5 of a more comprehensive model geared to the features of direct manipulation.

2.1 Elements of human-computer interaction

Human-computer interaction consists of a cycle of the user executing actions followed by an evaluation of the effects of those actions. According to Norman (1986) gulfs of execution and evaluation arise due to differences between a user's psychological representation of task related goals and the system's physical representations corresponding to these goals:

> [...] the person's goals are expressed in terms relevant to the person - in psychological terms - and the system's mechanisms and states are expressed in terms relative to it - in physical terms. The discrepancy between the psychological and physical variables creates the major issues that must be addressed in the design, analysis, and use of systems. I represent the discrepancies as two gulfs that must be bridged: the *Gulf of Execution* and the *Gulf of Evaluation.* (Norman, 1986, p. 39)

If the gulf of execution and the gulf of evaluation are easily overcome the interaction cycle will be characterised by what Bodker (1991) describes as smooth activity flow. This implies, to some extent, automatisation of the task in hand. Interruptions or unexpected results will break the activity flow, disrupting cognitive performance. As Norman (1991) states:

> The problem with disrupting activity flow is that the disruption brings to conscious awareness the disrupting activity, even when this is not the main focus of attention. This is usually undesirable, for it can have a negative impact on the task being performed. In fact, disruptions of this sort can lead to errors when the interrupting activity interferes with the maintenance of working memory for the task. The resulting memory difficulties may mean that the interrupted task is not resumed properly, either by being delayed beyond its proper execution time, by returning to the wrong point in the task, or by being forgotten altogether and never resumed: three classic forms of execution errors. (Norman, 1991, p. 24)

Norman refers to bridging the gulf of execution as a four segment process: intention formation, action specification, action execution, and making contact with the input mechanisms of the interface. Forming an intention will go some way to bridging the gulf of execution through the requirement to use an interaction language. As Norman (1986) puts it "the interaction language demanded by the physical system comes to colour the thoughts of the person" (Norman, 1986, p. 39). He describes action specification in the following way:

> [...] specifying the action requires translating the psychological goals of the intention into the changes to be made to the physical variables actually under control of the system. This, in turn, requires following the mapping between the psychological intentions and the physical actions permitted on the mechanisms of the system, as well as mapping between the physical

mechanisms and the physical state variables, and between the physical state of the system and the psychological goals and intentions. (Norman, 1986, p. 39)

The step of executing a specified intention is the first physical action required to bridge the gulf of execution. The gulf of evaluation is also bridged in four segments, a process which Norman describes as "starting with the output displays of the interface, moving to the perceptual processing of those displays, to its interpretation, and finally, to the evaluation - the comparison of the interpretation of system state with the original goals and intention" (Norman, 1986, p. 40-41). Considering the gulfs of execution and evaluation led Norman to propose seven stages in the execution-evaluation cycle as shown in Figure 2.1.

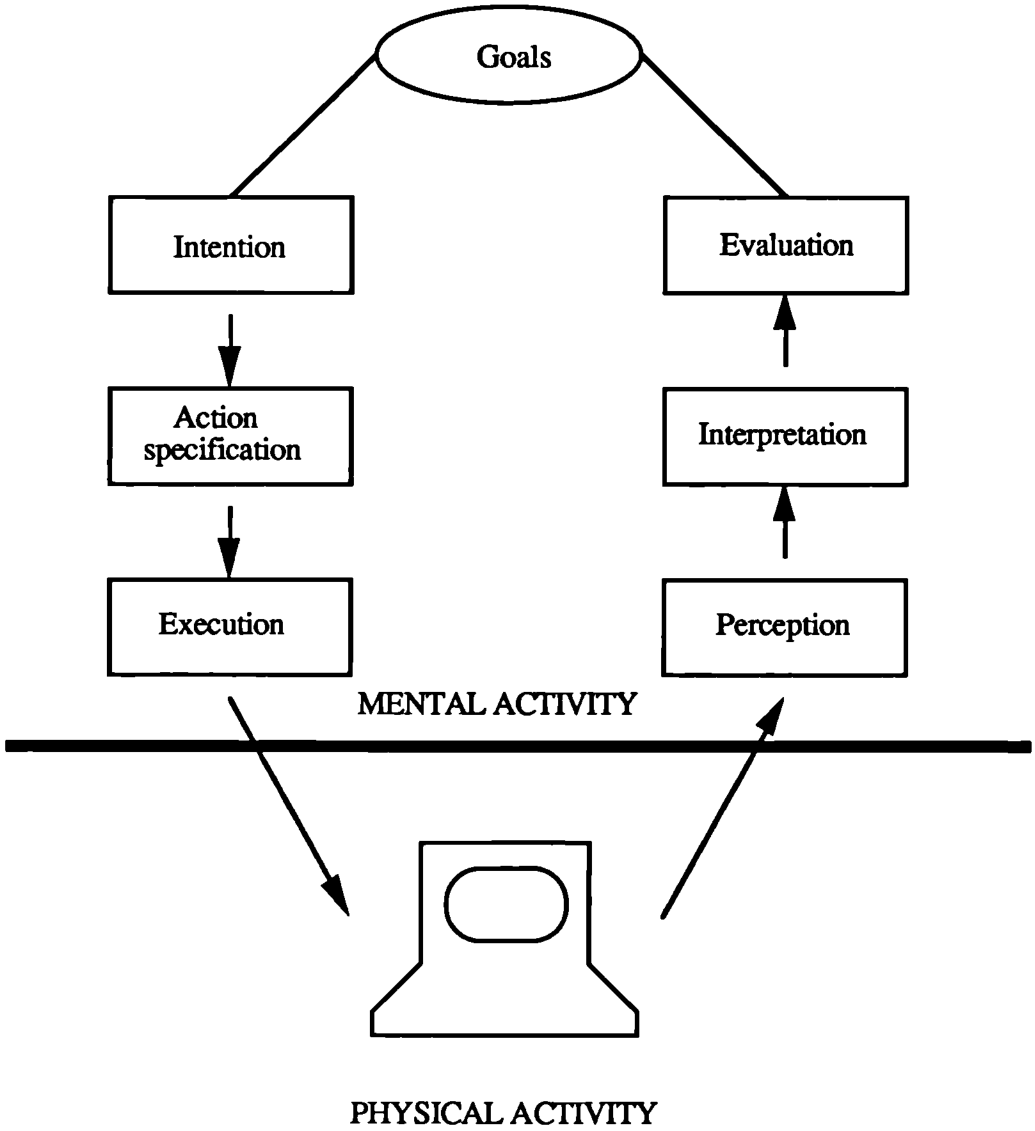


Figure 2.1: States in the execution-evaluation cycle as proposed by Norman (1986)

These stages are establishing the goal, forming the intention, specifying the action sequence, executing the action, perceiving the system state, interpreting the state, and evaluating the system state with respect to the goals and intentions.

Figure 2.1 illustrates that Norman considers that the stages of execution and perception mediate between psychological and physical representations. Likewise the input mechanism and the output displays also mediate between psychological and physical representations. This leads to the notion of a system image, that is the physical image of the system that users employ to develop their conceptual models. The user's interpretation of the system image is crucial in the development of a mental model of the system. A good system image will assist the user to develop mental models that are compatible with the design model of the system.

Abowd & Beale (1991) have extended the framework proposed by Norman. The execution/evaluation cycle as represented by Norman considers the system only as it is physically represented, that is by its system image. Abowd & Beale include explicit consideration of the system state, leading to a framework as shown in Figure 2.2. The framework is defined by Abowd & Beale as follows:

> The nodes represent the four major components in an interactive system - the *System*, the *User*, the *Input* and the *Output*. Each component has its own language which is used to express its purpose in the interaction. In addition to the *User's* task language and the *System's* core language [...], there are languages for both the *Input* and *Output* components to represent those separate, though possibly overlapping components. *Input* and *Output* together form the *system interface*. Note that we distinguish between the system image and the *physical interface*. The physical interface is that part of the system which, as its name suggests, is in direct contact with the user in the physical world. Therefore, the physical interface is viewed as a subset of the interface in our framework. The input and output languages do not in most cases map very directly onto the concepts in the domain. Yet, the interface's position between the *System* and the *User* mandates that it be an effective mediator for the tasks in the domain of application. (Abowd and Beale, 1991, p. 75)

Four steps constitute the interactive cycle between a user and the interface. Firstly, a goal and associated task are formed and articulated through the input language. Secondly, the input language is translated into the system or core language to define actions to be executed by the system. This results in a change in the system state corresponding to the input specification. The execution part of the cycle is now complete. Thirdly, the state of the system as represented by the core language is expressed in terms of the output language. Finally, the user observes the output to assess the results of the output relative to the original goal.

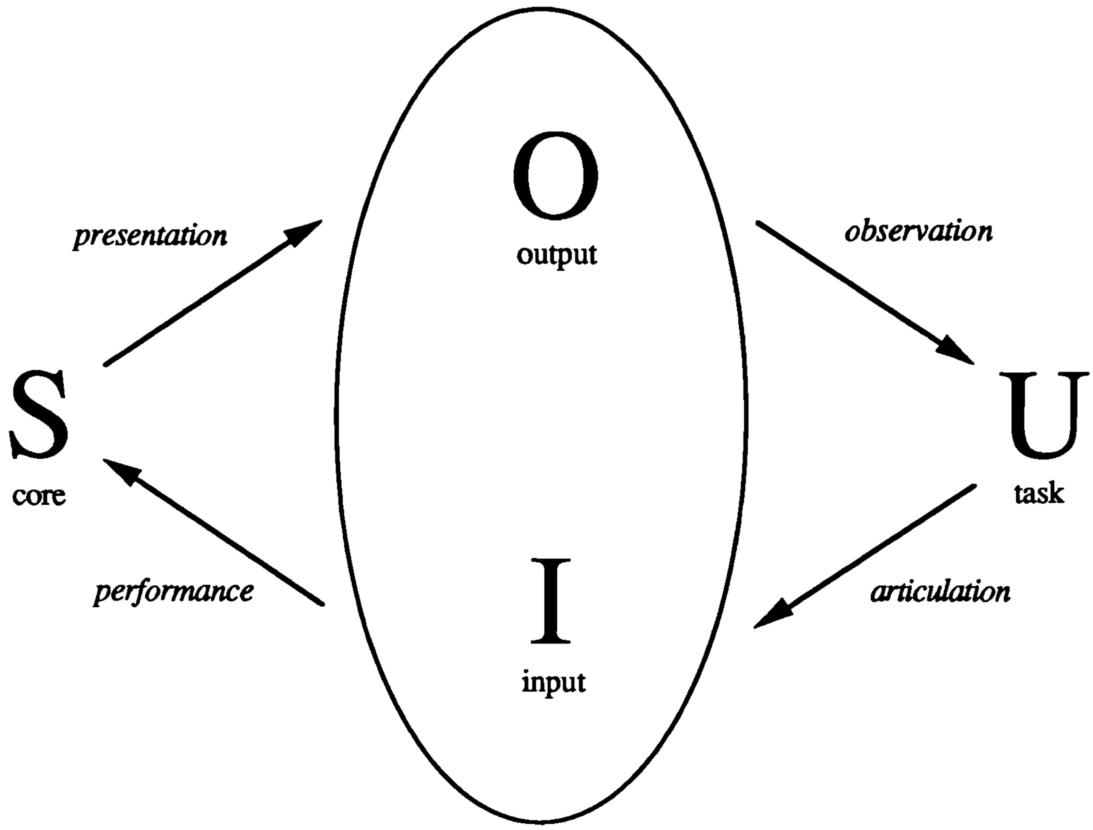


Figure 2.2: Human-computer interaction framework proposed by Abowd & Beale (1991)

Abowd & Beale invoke the notion of "distance", first used by Hutchins, Hollins & Norman (1985), as a qualitative measure of the way that the translation from each node effects the human-computer interaction. A reduction in the distance between any two nodes is seen as an improvement in design. For a perfectly designed system the distances would be zero. A consideration of their framework suggests that four types of distance are important: articulatory distance, functional distance, expressive distance, and observational distance. In addition, the notion of semantic distance is introduced as an aid to considering how easy it is to match the results of an interaction and the original goal.

Semantic distance is described by Abowd & Beale as the most important measure that can be applied to an interactive system. It is essential that the user is able to compare the results of an interaction with the original goal, implying a comparison between the semantics of intention and the semantics of evaluation. The difference between these two semantic descriptions is defined by Abowd & Beale as the semantic distance. A small semantic distance implies a close match between the goal structure of the user and the representation of this structure by the system. Payne (1987) expressed this as the yoked state space hypothesis: "The user of any device must construct and maintain two separate state spaces: the goal space, and the device space, and a mapping between the two" (Payne, 1987, p. 204). Thus for a small semantic distance the goal and device spaces must be "yoked", that is there should be a well defined relationship between these spaces so that tasks can be translated into device space transformations, and device states can be compared with the user's goals.

The notion of an "assimilation paradox" (Carroll & Rosson, 1987) is relevant to the semantic distance. This paradox refers to the way in which prior experience and learning can both facilitate new learning and hamper it; previous ideas and concepts may provide a basis for new learning, but they may also encourage erroneous or irrelevant assumptions about the new learning task. The approaches that Carroll & Rosson propose for coping with the assimilation paradox illustrate in a more general sense how the semantic distance can be reduced. Their first approach consists of attacking the paradox by providing the user with alternate mental models to the models based on previous experience which they possess. Halasz & Moran (1982) propose that this approach implies that users should be given accurate and complete explicit models of the system; an approach illustrated by du Boulay, O'Shea & Monk (1981), Moran (1981), and Young (1981). The second approach attempts to mitigate the effects of the paradox by making the system as similar to the user's prior experience as possible. Direct manipulation is a classic example of the approach. Another approach is to include naive expectations about performance in system design. The final approach is to design for the paradox and exploit the accommodation that users make when assimilation fails.

Articulatory distance refers to how easy it is for users to formulate their goals in the input language. A small articulatory distance implies a mapping between psychological entities in the task language and the input language. Observational distance is concerned with the ease and accuracy with which it is possible to translate the output language back into the task language. Possible approaches to minimising both the articulatory and observational distances are illustrated by a consideration of the "production paradox" (Carroll & Rosson, 1987). For novices the production paradox manifests itself in the way new users dismiss the use of manuals and on-line tutorials, preferring an exploratory approach, and for experienced users it is evident in the way they resist replacing new methods with tried and tested methods that they are happy with. As with the assimilation paradox Carroll & Rosson suggest attack, mitigation, and design as approaches to the production paradox. Attack involves making the use of the interface intrinsically rewarding; rather like playing a game and thus epitomising the positive motivational features of curiosity, motivation and challenge identified by Malone (1981). Mitigation can be effected by controlling the consequences of an operation (for example, the provision of undo operations or the use of reversible direct manipulation procedures) or by controlling the options available to the user (hiding, dimming, and blocking options). The design approach is based on the notion of making the design of the systems and documentation more task orientated.

Functional distance is concerned with the extent to which the core language representation of the input language can reach as many states of the system as possible. The more states of the system that can be reached the smaller the functional distance. Expressive distance refers to the extent to which it is possible to translate the core language into the output language while preserving the system attributes relevant to the task in hand. It is critical that the differences in the system state at the start and finish of the execution phase of the interaction cycle are adequately represented in the output language.

In an easy to use system the interface will be close to the user, that is the articulatory and observational distances will be small. Abowd & Beale (1991) define the distance between the user and the interface as the user perceived difficulty. Hutchins et al. (1985, 1986) use the term semantic distance in a different way from Abowd and Beale to describe this measure. The complementary distance is the system perceived difficulty. This is different from the functional and expressive distances. These two distances refer to the possible states that the system can be in, while the system perceived difficulty refers to the ease with which state changes can take place. Interface design involves a compromise between coping withthe demands of reducing the user perceived difficulty and the system perceived difficulty. Typically as the user perceived difficulty is reduced the system perceived difficulty is increased, placing greater demands on the designer.

The most commonly advocated way to reduce user perceived difficulty is to adopt a consistent design approach (see, for example Reisner, 1981; Rubinstein & Hersh, 1984; Payne & Green, 1986; Shneiderman, 1987; Kellogg, 1987). Dix, Finlay, Abowd & Beale (1993) aptly describe the high status afforded to consistency:

> Consistency is probably the most widely mentioned principle in the literature on user interface design. "Be consistent!" we are constantly urged. The user relies on a consistent interface. However, the difficulty of dealing with consistency is that it takes many forms. Consistency is not a single property of an interactive system that is either satisfied or not satisfied. Instead, consistency must be applied relative to something. (Dix et al., 1993, p. 135-136)

However, although consistency is a much prized attribute, there is considerable debate to exactly what constitutes a consistent interface. Reisner (1993) summarised the lack of a clear definition as follows:

> [...] precisely what is meant by this term [consistency] remains illusive. Loosely, consistency has been described as "doing similar things in similar ways", but throughout the years a number of authors have lamented our lack of a clear understanding of the term [...] (Reisner, 1993, p. 215)

As Dix et al. (1993) point, out consistency must be applied relative to something, and this requirement seems to be at the root of the confusion over the meaning of consistency. Grudin (1989) provides a thee fold categorisation of consistency in an attempt to clarify the situation: (i) internal consistency of the design with itself, (ii) external consistency with other interface designs, and (iii) external analogic or metaphoric correspondence of the design to the world beyond the computer domain. A very common analogy that is used as a basis for consistency is the system itself; an approach which Grudin (1989) sees as being problematical:

> A special case of designing by analogy is that of designing the user interface to correspond to the underlying system architecture [...]. The system architecture is external to the user interface and will not be familiar to many users, but is typically very familiar to the designers. Although it may work against the user, mapping the system architecture onto the user's interface is very seductive, appealing to the designer's sense of consistency and simplicity. [...] But as our designs adapt to the way that communities actually use systems, it is more likely that the user interface and system design will diverge, *reducing* the appropriateness of burdening the user with both descriptions. (Grudin, 1989, p. 1171)

System based consistency may reduce the system perceived difficulty, but it will probably increase the semantic distance and user perceived difficulty. As Grudin point out, a task based consistency may be a more appropriate design principle.

2.2 The direct manipulation paradigm

Hutchins et al. (1985) state that " [...] "direct manipulation" is not a unitary concept, nor even something that can be quantified in itself. It is an orienting notion. "Directness" is an impression or feeling about an interface" (Hutchins et al., 1985, p. 317). This rather intuitive interpretation of the direct manipulation is evident in the descriptions offered by other writers. Rutkowski (1982) claims that the user is able to apply intellect directly to the task, with the tool itself seeming to disappear. Nelson (1980) invokes the "principle of virtuality", that is the use of a representation of reality that can be manipulated, as an explanation. In Section 2.2.1 a more rigorous interpretation of the notion of direct manipulation is attempted with reference to the Abowd & Beale framework.

There are many reports of positive feelings about direct manipulation interfaces such as spreadsheets, display baseed editors, video games, and desk-top interfaces. Users claim mastery of the system, competence in performance of the task, ease in learning the system originally and in assimilating advanced features, confidence in the capacity to retain mastery over time, enjoyment in using the system, eagerness to show the system to novices, and desire to explore more powerful aspects of the system. Section 2.2.2 provides a review of empirical studies concerned with the efficacy of direct manipulation interfaces.

2.2.1 Human-computer interaction in direct manipulation interfaces

"Directness" provides the basic rationale for the design of direct manipulation interfaces. It is assumed that directly manipulating representations of objects relevant to the task in hand will provide an input language which is intuitive and easy to understand The output language should also be direct, as emphasised by Hutchins et al. (1985):

> "[...] the output language must represent its subject of discourse in a way that natural language does not normally do. The expressions of a direct manipulation output language must behave in such a way that the user can assume that they, in some sense, *are* the things they refer to. (Hutchins et al., 1985, p. 319)

Hutchins et al. (1985, 1986) introduce the idea of the model world metaphor to discuss the notion of directness. In the model world metaphor the interface actually provides the environment for interaction; an environment in which the user can act and the effects of the user's actions are explicitly represented by changes in the state of the interface. The user has a sense of direct engagement and there is no intermediary between the user and the system. The user is encouraged to think of the interface as the "universe" of their interaction, a feeling that diSessa (1985) refers to as "naive realism". This contrasts strongly with the conversation metaphor for interfaces, typified by command driven programs. Interfaces based on this metaphor support interaction through the intermediary of an interface, consisting of a language medium, which translates user intentions into system instructions. The user and the system have a conversation about an assumed but not explicitly stated world.

The directness of an interface will depend not only on the design of the system but also on the characteristics of the task and the user, that is the user perceived difficulty. This in turn implies that both the articulatory and the observational distances should be small. As with all well designed software the semantic distance should also be small. These criteria are discussed with reference to three types of types of directness - semantic, operational and formal directness (Ziegler & Fahnrich, 1988) .

Semantic directness depends on the ease with which the user can (i) execute and observe the effects of actions, that is the sizes of the articulatory and observational distances, and (ii) compare the expression in the task language of the original goal with the expression in the output language of the result of an interaction, that is, the magnitude of the semantic distance.

The articulatory and observational distances will be reduced by matching the nature of the interface objects to the nature of the task. For example, the objects in the Macintosh interface mimic the tasks which the user is expected to perform - a waste paper basket icon is used to delete files, multiple documents are represented as overlapping sheets of paper, and so on. The visual objects used in the interface should afford the user a visualisation of the task which is compatible with the mental model of the task held by the user. As Shneiderman points out this is not a trivial task:

> Choosing the right objects and actions is not an easy task. Simple metaphors, analogies, or models with a minimal set of concepts seem most appropriate to start. Mixing metaphors from two sources may add complexity which contributes to confusion" (Shneiderman 1987 p. 201)

The articulatory distance also depends on the number of operations that a user needs to execute in order to complete a task. As Ziegler & Fahnrich (1988) point out the need to directly manipulate multiple objects or to execute a sequence of operations is undesirable:

> Semantic directness is low if in order to work with one conceptually-coherent object of a task domain several different objects have to be manipulated or composed. Semantic directness is also low if a certain intended operation can be achieved only by a sequence of system operations. (Ziegler & Fahnrich, 1988, p. 127)

Direct manipulation is intended to make the semantic distance small by making the interface objects both input and output devices, a property of interfaces that Draper (1986) terms *inter-referential* input and output. In this way the semantics of goal expression expressed through the input language and the semantics of evaluation expressed through the output language will refer to the same medium, increasing the chance of a match between the two semantic expressions. Hutchins et al. emphasise this point:

> [...] the output language must present representations of objects in forms that behave in the way that the user thinks of the objects behaving. Whatever changes are caused in the objects by the set of operations must be depicted in the representation of the objects. This use of the same object as both an input and an output entity is essential to providing objects that behave as if they are the real thing. It is because an input expression can contain a previous output expression that the user feels the output expression is the thing itself and that the operation is applied to the thing itself. (Hutchins et al., 1985, p. 333)

Operational directness also relates to articulatory distance and observational distance. Users should be able to execute the actions that they feel are necessary; if users are excessively constrained in their choice of actions by the system design the articulatory distance will be large. If they do not receive prompt and accurate feedback the observational distance will be large. The need for operational directness leads to commonly accepted design guidelines, as Ziegler & Fahnrich point out:

> It is usually difficult to make valid assumptions about the user's intended action sequences. Most DM systems therefore try to make all visible objects accessible in a user-initiated manner and to support the evaluation of a sequence

> of actions by immediate feedback after each step. (Ziegler & Fahnrich, 1988, p. 127)

Formal directness refers to "the immediate comprehensibility of the system's output and an easy and efficient handling of the input elements and devices (commands, buttons, mouse handling etc.)" (Ziegler & Fahnrich, 1988, p. 127). As such formal directness affects both the articulatory and the observational distances. The use of icons, object and function selections instead of command names, well designed screen layout, and the comprehensible naming of function keys all contribute to formal directness. Typically these objects are represented pictorially, rather than textually or numerically. As Shneiderman (1987) points out physical, spatial, or visual representations appear to be easier to retain and manipulate than do textual or numeric representations. The classic approach to achieving formal directness is "what you see is what you get" (WYSIWYG), in which documents are shown on the screen in exactly the same form as they will appear in print-out.

It is clear that the implications of directness are consistent with the three original design principles stated by Shneiderman (1982, 1983) for direct manipulation: (i) continuous representation of the objects and actions of interest, (ii) physical actions or labelled button presses instead of complex syntax, and (iii) rapid incremental reversible operations whose impact on the object of interest is immediately visible. Hutchins et al. (1985) provide a more detailed set of requirements:

- Execution and evaluation should exhibit both semantic and articulatory directness.
- Input and output languages should be inter-referential, allowing an input expression to incorporate or make use of a previous output expression. This is crucial for creating the illusion that one is directly manipulating the objects of concern.
- The system should be responsive, with no delays between execution and the results, except where those delays are appropriate for the knowledge domain itself.
- The interface should be unobtrusive, not interfering or intruding. If he interface itself is noticed, then it stands in a third-person relationship to the objects of interest, and detracts from the directness of the engagement.

(Hutchins et al., 1985, p. 333)

While it appears that direct manipulation may make some tasks easier there are some tasks which are probably best done by using other methods, for example repetitive operations are probably best described in terms of a script. Other problematic applications of direct manipulation include handling variables, distinguishing depiction of an individual element from a representation of a class of elements, and precision in specification by the user.

A more fundamental problem relates to the fact that as the user adapts to the system representation they may change their own conceptualisation of the problem so that they come to think of it in the same terms as the system. Hutchins et al. make this point:

> "While moving the interface closer to the users' intentions may make it difficult to realise some intentions, changing the users' conception of the domain may prevent some intentions from arising at all. So while a well-designed special purpose language may give the users a powerful way of thinking about the domain, it may restrict the users' flexibility to think about the domain in different ways." (Hutchins et al. 1985, p. 328)

As direct manipulation interfaces are built on ideas and concepts that users already possess, this may restrict users in using computational environments to think about things in new ways.

2.2.2 Empirical studies of the use of direct manipulation interfaces

Despite the extensive claims made for use of direct manipulation there have been remarkably few empirical studies of the use of direct manipulation.

Whiteside, Jones, Levy & Wixon (1985) conducted a comparison of the usability of seven interfaces, two of which involved direct manipulation. This study found no general advantages associated with the use of direct manipulation. However, the scope of the tasks (standardised filing tasks) and the short time span of this study indicate that this experiment should not be regarded as a complete evaluation of direct manipulation.

More recent studies have demonstrated positive advantages linked to the use of direct manipulation. Ziegler & Fahnrich (1988) report that Frese, Schulte-Gocking & Altmann (1987) compared learning and performance with a direct manipulation word-processor (*MacWrite*) and a command driven word-processor (*Wordstar*). Little difference in user performance during the initial introductory sessions was found, but that the direct manipulation system was superior in succeeding sessions. It was also better for more complex tasks. They also report that Shneiderman & Margono (1987) compared simple file manipulation tasks completed in direct manipulation and command driven interfaces. The results of this study indicated that direct manipulation was better with respect to task completion times, subjective rating by the user, and ease of learning. A comparison by Eberts & Bittianda (1993) of complex file manipulation tasks completed in direct manipulation and command driven interfaces led to less conclusive results. They found a preference for direct manipulation if a task could be conceived in concrete spatial terms, but no clear preference when the task was abstract.

2.3 Models for human-computer interaction

Human-computer interaction is typically complex, involving a consideration of the interaction between system design, task requirements and users' physical and mental capabilities. This inherent complexity implies that models of human-computer interaction will also be complex. To cope with this potential complexity, decisions about what features of human-computer interaction to attend to, and what features by default not to consider, are typically made in the design of interaction models. This has led to the development of a large number of models of different types, with strengths and weaknesses associated with each type of model. As Simon (1988) states:

> Once one introduces the complexity of the human and cognitive and physical systems and then tries to estimate the effect of their interaction with some (more or less) pragmatically designed device, the resulting predictions will necessarily be of limited breadth and depth in coverage. [...] since any [user] model can only ever hope to account for a part of the massive complexity of human behaviour in any context, the diverse approaches which exist are the result of pragmatic decisions about what function any model which is developed will serve [...]. The resulting state is that any would-be user of cognitive models faces a confusing array of partial tools when considering which approach to adopt. (Simon, 1988, p. 80)

A common distinction is made between competence and performance models (for example Green, Schiele & Payne, 1987). Simon (1987) describes the difference between these types of model as follows:

> Competence models tend to be ones that can predict legal behaviour sequences but generally do this without reference to whether they could actually be executed by users. In contrast, performance models not only describe what the necessary behaviour sequences are but usually describe both what the user needs to know and how this is employed in actual task execution. (Simon, 1987, p. 81)

Dix et al. (1993) have categorised models in a different way as (i) goal and task hierarchies, (ii) linguistic models, (iii) physical and device models, and (iv) cognitive architectures. This categorisation will be used to review established models.

2.3.1 Goal and task hierarchies

Goal and task hierarchies assume that human behaviour can be described in terms of hierarchically related goals which are attempted by the user in a sequential fashion. This assumption has been criticised by some writers. For example, Whiteside & Wixon (1987) do not think that users start with goals at all, "rather they are always already acting in a situation, thrown to it as it were, unreflectively and unanalytically" (Whiteside & Wixon, 1987, p. 360), and Hayes-Roth & Hayes-Roth (1979) advocate that planning is often an opportunistic affair. However, the following claim by Black, Kay & Soloway (1987) indicates the conviction of some writers in the validity of goal based behaviour:

> Almost all of human behaviour can be characterised in terms of goals and plans. In particular, most of what people do is devise plans of action and perform them in order to bring about some desired state of the world - that is, to accomplish a goal. Consequently, much of human knowledge about how to operate in the world is stored in memory in the form of plan and goal knowledge representations. (Black, Kay & Soloway, 1987, p. 36).

A belief in goal based behaviour is also evident in the development of the well known ACT* theory of skill acquisition (Anderson, 1982, 1983, 1987). Anderson described the critical role of a hierarchical goal structure in ACT as follows:

> The ACT* production system specifies a hierarchical goal structure that organises the problem solving. [...] It has proven impossible to develop satisfactory cognitive models that do not have some overall sense of direction in their behaviour. As can be seen with respect to this example [writing the FIRST function in Lisp], the hierarchical goal structure closely reflects the hierarchical structure of the problem. (Anderson, 1987, p. 196)

In a goal and task hierarchy the user's model is represented in terms of a set of "top-level" goals, each of which can be represented by hierarchically arranged sub-goals. Top-level goals consist of relatively modest well-defined tasks. The lowest level goals are associated with "unit-tasks" that do not need the user to exercise any problem solving skills. The underlying assumptions of these models make them most applicable in "routine" contexts, that is contexts in which human-computer interaction can be described in terms of simple well defined-and clearly related goals, that can be achieved by the serial execution of limited tasks. Their application in contexts involving parallel processing, extensive problem solving and frequent user errors is problematical.

The best known examples of goal and task hierarchies belong to the Goals Operators Methods Selection (GOMS) family of models. The original GOMS model is described by Card, Moran & Newell (1980a) and Card, Moran & Newell (1983). In this model a goal is described by Card et al. as "a symbolic structure which defines a state of affairs to be achieved and determines a set of possible methods by which it may be accomplished" (Card et al., 1983, p. 144). Operators are defined as "elementary perceptual, motor, or cognitive acts, whose execution is necessary to change any aspect of the users mental state or to affect the task environment". (Card et al., 1983, p. 144). A method gives a procedure for accomplishing a goal, and is described as a sequence of goals and operators, with conditional tests on the contents of the user's immediate memory and on the state of the task environment. Selection is the decision by the user on which method to use.

Cognitive Complexity Theory (CCT) has been developed as an extension of the GOMS model (Kieras & Polson 1985; Bovair, Kieras & Polson 1990), which incorporates parallel descriptions of the user and the system. The user's knowledge of the system is represented by an extended version of GOMS, with production rules included to represent the cognitive behaviour of the user:

> The relationship between the GOMS model and the production system formalism is direct. Goals are directly represented in the production system for job-task knowledge. They appear in the conditions of almost all production rules, and are manipulated in the many production actions. Methods appear in the form of sequences of production rules whose first member is triggered by the assertion of the goal of the method. The structure of sub-goals incorporated in a method appears in the form of actions in the method productions that assert and delete goals. Selection rules are simply the production rules that control the execution of methods. Thus, a selection rule can be a single production rule that asserts the goal that triggers the execution of a method, or it can be a collection of productions evaluating the current context in order to select the best method for that context. Operations are also scattered through the whole system, and consist of elementary actions, as well as environment-testing conditions. (Kieras & Polson, 1985, p. 369-370).

The device/system representation is given in terms of Generalised Transition Networks (GTN), a method description developed by Kieras & Polson which they describe as follows:

> [...] a GTN consists of nodes, which represent states, interconnected by arcs, which represent possible transitions between states. In diagram form, the nodes are represented by circles, and the arcs are represented by arrows between the circles. An arc consists of a condition, an action, and a specified next state. The arcs appear in a specified order; in the diagrams, this order is clockwise round the circle that represents the state, beginning at the top. The condition is written above the arc, the action is written below the arc. The next state is simply the circle to which the arrowhead points. (Kieras & Polson, 1985, p. 381-382)

As with the original GOMS model only routine cognitive performance is modelled; problem solving, for example, comprehending a manuscript, is not modelled by CCT, as made clear by Bovair et al. (1990):

> In the cognitive complexity approach, the production system model of the user makes no attempt to represent fundamental cognitive processes [...]. Such complex unanalysed processes are represented by the appearance of special operators in the rule actions [..] (Bovair et al., 1990, p. 8).

However, there are two significant differences between the GOMS and CCT approaches. Firstly, in CCT the structure, extent, and complexity of task related knowledge can be computed by calculating the number of production rules in a method and the level to which goals/sub-goals are nested. In the GOMS formalism only the content and structure of knowledge associated with a task is represented. Secondly, the inclusion of GTNs in CCT allows a comparison between the user's knowledge of a system and a formal representation of the system, as illustrated by Kieras & Polson (1985) in their comparison of "goal tree" diagrams to represent the mapping between device hierarchies and users' goal hierarchies. GOMS does not include a principled consideration of system characteristics.

The strength of goal and task hierarchies is in the articulation phase in the Abowd & Beale interaction framework. The set of top-level goals and sub-goals provides a description of the user's cognitive model of the task involved and provides a basis for interpreting the actions that they perform to achieve these goals. These models are less successful in representing the observation phase, although there is some scope for the consideration and interpretation of output in terms of the user's goal structure. The GOMS formalism does not address the performance and presentation phases. However, the performance phase is considered in the CCT approach through the comparison of user goal hierarchies and the device goal hierarchies obtained from an inspection of GTNs.

2.3.2 Linguistic models

According to Simon (1988) linguistic models utilise the idea of generative grammars, an idea borrowed from linguistics, in which it is possible to generate all and only the legal sentences in a language. In human-computer interaction terms "sentences" correspond to sequences of actions, and the language corresponds to the interaction language between the user and the interface. An application of a linguistic model would consist of the use of grammar rules to express the set of possible user-executable actions. Green, Schiele & Payne (1988) provide a review of linguistic models of human computer interaction, ranging from the use of Backus-Naur Form (BNF) grammar to describe user actions at an interface reported by Reisner (1981) to the development of a task action grammar (TAG) described by Payne & Green (1983, 1986). Tauber (1988) and Howes & Payne (1990) report extended versions of TAG.

BNF grammar was developed to describe programming languages with the grammar taking the form of rules which will provide correct statements in the programming language. As such the grammar is purely syntactic, and its application to human-computer interaction takes no account of any semantic representation of the dialogue between the user and a computer. In this sense the BNF grammar model lacks a cognitive dimension; it only provides a description of what actions it is legally possible for the user to execute, without considering what leads to the decisions to take specific actions.

TAG attempts in two ways to provide a cognitive dimension to a linguistic interpretation of the user interface. Firstly, instead of rewriting actual machine commands an attempt is made to use grammar rules to express simple tasks that are assumed to be perceived by the user. Secondly, a TAG analysis aims to consider the consistency of an interface in terms of the alignment between syntactic and semantic aspects of the interface language:

> In the ideally consistent [interface] language, semantic relations will not only be mirrored in the lexical or symbolic relations, but also in the structure of the commands. [...] The task semantics should map onto the the language syntax in a consistent way. (Payne & Green, 1986, p. 96)

An approach including consistency as one foci is justified by Payne & Green (1986) as follows:

> Consistency [...] is informally recognised to be a major determinant of learnability. The advantages of consistency lie in facilitating generalisations by the user, who having learned some parts of the system can then infer others. (Payne & Green, 1986, p. 95)

In order to effect a measure of consistency simple complexity metrics are computed, with the supposition that the more complex an interface language is, the less consistent it is.

The extended task action grammar described by Tauber (1988) attempts to increase the cognitive dimension of the TAG model further. Tauber (1986a, 1986b) has introduced the notion of the "user's virtual machine" (UVM) to describe the user's perception of the structure of knowledge needed to understand the task related work of a system. The UVM thus provides a framework for linking a user's mental model of the interface with the ideal representation of competent performance predicted by formal grammars. The extension of TAG to represent display based systems has been attempted by Howes & Payne (1990) through the development of D-TAG.

2.3.3 Physical and device models

Physical and device models are designed to provide reasonably accurate, detailed descriptions of the performance of well defined unit tasks (for example, insertion of a character in a document using a word processor) at the sensory motor level. A well known example is the Keystroke Level Model (KLM) described by Card, Moran and Newell (1980b, 1983). A more contemporary example concerned with current input devices is the three state model for input devices (Buxton 1990).

The KLM is an application of the GOMS model at a the lowest level of analysis; the keystroke level. Card et al. (1983) give four levels of analysis for the GOMS model - analysis at the unit-task, functional, argument, and keystroke level. They describe each of these levels with respect to text editing. At the unit-task level the task goal is achieved by the repetition of an edit unit-task operator. In functional level models tasks are decomposed into the cycle of (i) get the next unit-task, (ii) locate the cursor at the edit site, (iii) make the required modification, and (iv) verify that the edit has been completed successfully. At the argument level the individual steps at the functional level are decomposed into the individual steps and specifying actions and their arguments. The operators used at the keystroke level are different than those used at the other levels. The concern is no longer with the functional role of the interface language, but rather the basic physical and mental actions of the user, such as typing, clicking a mouse button, looking, moving a hand, and various mental operations.

The strength of the KLM is in the prediction of times required to execute unit-tasks. This is done by describing the execution of a unit-task in terms of four different physical-motor operations: keystroking, pointing, homing and drawing; one mental operator, taken as pauses between the execution of physical operations; and a system response operator. Most operators are assumed to take a constant time for each occurrence. Actual execution times are obtained from empirical data. For example, the time to point with a mouse can be determined by applying Fitt's Law (Fitts & Posner, 1967) and thus depends on the size and position of the target. A representative value would be 1.10 seconds. The execution time for a unit-task is simply calculated by summing the individual execution times.

The KLM is a very basic "engineering" model which takes no account of cognitive factors. However, within these constraints it has proved to be very successful. Card et al. (1983) empirically validated the prediction times derived from an application of the KLM to the analysis of a wide range of tasks, finding an error of about 20%. However, these results should be viewed with caution, as the accuracy of the predictions will critically depend on the representative individual operator execution times that are used to calculate the total execution time.

The three state model has been specifically designed to address the use of modern input devices by looking at different pointing tasks, such as icon selection and line drawing, in order to assess the performances of various pointing devices in these contexts. It is designed to capture the differences between using commonly available input devices, such as a mouse, a light pen and a tracker ball. As the name of the model implies the state of a pointing device is defined in terms of three values. For example, consider a light pen with a button. State 1 corresponds to the light pen touching the screen without the button depressed, State 2 corresponds to the pen touching the screen with the button depressed, and State 0 occurs when the pen is not touching the screen A mouse has two states - State 1 when it is being moved across the screen and State 2 when it is being dragged across the screen. The application of a device to a specific task is described in terms of the required 0-1-2 state changes.

2.3.4 Cognitive architectures

The models previously considered have implicit or explicit representations of cognitive processing, even if these representations are at a superficial level. For example, the GOMS model uses a hierarchical goal directed description of behaviour, CCT assumes a distinction between short and long term memory, with production rules stored in long term memory which are invoked when they match the contents of short term memory, and the KLM is based on the Model Human Processor cognitive architecture (Card et al. 1983). In this section the problem space model and the interacting cognitive sub-systems (ICS) models are described a examples of models in which the associated cognitive architecture is central to the model.

The problem space model, originating from the work of Newell & Simon (1972), defines a problem space which is traversed by making plans to reach a goal from a given state. A problem space consists of a set of states and operations which can be executed to change from one state to another state. The desired goal is represented by a sub-set of the possible states. To achieve this goal operations are executed to change the existing state of the problem space so as to correspond to this sub-set. This model is interesting in that it can be used to model exploratory non-routine cognitive behaviour, in contrast to the previously described models.

The problem spaces model is not directly implementable. However, it has given rise to executable cognitive architectures such as Soar (Laird, Newell & Rosenbloom 1987). Soar has in turn been used to develop programmable user models (PUMs) as described by Young & Green (1989). PUMs aim to provide a programmable cognitive architecture which can be used to simulate the use of an interface design specified in terms of knowledge analysis of the task in hand. The simulation is intended to represent user behaviour itself, and as the description of the interfaces is not given in parametric or idealised terms, it will not provide predictions of task completion times or estimates of system complexity.

Interacting cognitive sub-systems (Barnard, 1985; Barnard, 1987; Barnard, Wilson & Maclean, 1988; Barnard & Teasdale, 1991; Teasdale & Barnard, 1993; Barnard & May, 1993) is a model which provides a model of perception, cognition and action. Barnard (1987) explains the basic assumptions of ICS s follows:

> It assumes that perception, cognition, and action can usefully be analysed in terms of well defined and discrete information processing modules; it assumes a separation of mental representations stored in episodic memory records from the content-specific processes that construct and utilise them; and it assumes a significant capability on the part of the human cognitive system for parallel processing of information. (Barnard, 1987, p. 20)

Thus ICS is built up by the coordinated activity of nine sub-systems - five peripheral; systems in contact with the world, and four concerned with mental processes. The model is not intended to provide a description of the user in terms of sequences of actions, rather it is intended to provide a holistic view of the user as an information processing machine. As Simon (1987) remarks when comparing ICS to the MHP model of Card et al.:

> In contrast, ICS represents the nature of processing more strongly and in a less parameterised fashion. It is able, therefore, to capture wider variations in the behaviour due to performative factors, including some classes of errors. As a result the interaction between cognitive resources available to the user and those required by the task should be more adequately covered by ICS than MHP, for example. (Simon, 1987, p. 89)

2.4 Problems with applying existing models to direct manipulation interfaces

Most established usable models of human computer interaction were developed before the use of direct manipulation interfaces became common. Thus most of the models have been framed in terms of command driven interfaces, or at best keyboard and cursor based interfaces.

The major problem that both goal hierarchy and grammar methods have in representing direct manipulation is their concentration on input aspects with a failure to deal adequately with the output features. There is an implicit assumption that the users know what they want to do and that feedback from the output is used simply to display the results of their actions rather than provide a clue for future interaction with the system. This almost amounts to heresy in a direct manipulation context , where a fundamental principle is an inter-referential relationship between input and output. Grammars also face a problem in determining the appropriate resolution of analysis that should be adopted. Dix et al. (1993) illustrate this point when they state that pressing a cursor key is a reasonable lexeme, while moving a mouse pointer one pixel makes less sense. In addition, the meaning associated with clicking a mouse pointer at a given screen position depends on the current state of the display. The application of structured methods as a basis of goal hierarchies also presents problems. In direct manipulation systems actions are easy to execute and undo, leading to a sense of security which encourages an exploratory approach to task completion. There is obvious scope for a mismatch between the open style of interaction sponsored by direct manipulation and closed representation afforded by GOMS type models.

From the above it is clear that an ideally applicable established model to represent direct manipulation does not exist. Consequently it has been decided to select one of these models as a basis for an initial representation of the direct manipulation observed during the laboratory sessions featured in this research programme, with view to expanding this model in light of the research findings. The choice of an initial model has been made with respect to four criteria. Firstly, there should be evidence that the method has not atrophied, but that it is capable of being developed and expanded. Secondly, reasonably successful attempts should have been made to apply the model in diverse contexts. Thirdly, it should be possible to apply the method in a relatively straightforward and realistic fashion. Fourthly, the representation afforded by the model should at least in some senses match the form of interaction envisaged during the research programme.

The GOMS approach has provided a basis for a number of extensions. The development of CCT was an expansion of the GOMS model to incorporate a description of the system and an enhanced representation of the user. The credibility of the CCT extension is demonstrated by the capability of the extension to predict the transfer of training from one system to another (Bovair et al., 1990; Kieras, 1988; Kieras & Bovair, 1986; Kieras & Polson, 1985; Polson & Kieras, 1985; Polson, Muncher & Englebeck 1986). John (1988) and John & Newell (1989) have also expanded the GOMS model by adding techniques borrowed from critical path analysis to include a consideration of parallel processing. Olson & Olson 1990) review research by Lerch (1988), Lerch, Mantei & Olson (1989), and Smelcer (1988) on the analysis of errors in a GOMS representation to argue that the analyses presented in these research studies "only open the door on the treatment of errors within the GOMS framework, but the work described offers a significant beginning toward addressing one of the major shortcomimgs of the GOMS model" (Olson & Olson, 1990, p. 250-251).

The original and best known application of GOMS is in the text editing domain (see for example Card et al., 1983; Polson, 1987; Singly & Anderson, 1988). However, Gugerty (1993) states that the GOMS approach has been adopted in a number of other task domains, including editors for tables and graphics (Bennett, Lorch, Kieras & Polson, 1987; Vossen, Sitter & Ziegler, 1987), telephone operator workstations (John, 1990), microprocessor-based oscilloscopes (Lee, Polson & Bailey, 1989), operating system commands (Karat, Fowler & Gravelle, 1987; Karat & Bennett, 1989), help systems (Elkerton & Palmiter, 1991), spreadsheets (Olson & Nilsen, 1988; Lerch, Mantei & Olson, 1989), and databases (Smelcer, 1988). It is interesting to note that this list of applications includes two research studies of the application of the GOMS model to the use of spreadsheet, a software application based on the direct manipulation paradigm.

Kieras (1988) acknowledges that there are problems in the application of the CCT extension of the GOMS approach. He identifies these problems as (i) difficulty in constructing production rule simulation models, and (ii) difficulty of doing, in a standardised reliable way, the detailed task analysis required to construct the representation of the user's procedural knowledge of how to operate the system. In response to these difficulties Kieras developed a language called Natural GOMS Language (NGOMSL). He reports that it is possible to use this language to conduct and use GOMS task analyses in a standard way which could be taught to people who were not specialists in human-computer interaction or cognitive psychology.

As reported in Chapter 6 an inspection of the specific tasks that research subjects were to be asked to complete indicated that these tasks could be expressed in terms of a hierarchical goal based structure. In turn these goals could be related to the execution of methods consisting of the completion of unit-tasks through the execution of a limited set of well defined operators.

The GOMS approach appears to fit the four criteria outlined above. Consequently the GOMS model has been adopted as the basis for developing a more comprehensive evaluation model for educational software.

2.5 A comprehensive model for educational software evaluation

The basic GOMS model provides a prescriptive competence model of the user, that is it can be used to predict legal methods without reference to whether users will adopt these methods. Thus the GOMS approach can be used to match observed behaviour with predicted behaviour, providing a way of comparing learner and designer models. However, it will not provide an explanation for observed behaviour, and thus it will not provide a description of the differences between learners' models and the designer's model. Hence, GOMS is good at pointing out the existence of design problems, not in providing an explanation of them.

The GOMS approach is also limited primarily to the input phase of the interaction cycle. It provides a representation of how users articulate task related actions, but very little attention is paid to how users interpret the results of their actions, and no consideration is given to the users' perceptions of the system.

The limited scope of the GOMS approach is consistent with the distinction drawn by Lewis (1991) between the application of "inner" and "outer" psychological theory:

> The kind of psychological theory stressed by Card et al. (1983) can be called *outer* theory: It describes what mental processes do, and how they are influenced by external factors, but does not describe how the processes are accomplished. By contrast, an *inner* theory, describes the mechanisms underlying the processes, ideally in such a way that an outer theory could be deduced from the description. (Lewis, 1991, p. 154)

This suggests that in order to develop an explanatory evaluation model for educational software the model must incorporate an inner theory of learning. In addition, the theories of user interaction developed by Norman (1986) and Abowd & Beale (1991) stress the importance of the user's perception of the system, suggesting that a system representation should be included.

A combination of an outer psychological theory, an inner psychological theory and a system representation implies a complex model. This is not surprising as it is a truism that learning is complex, and adding a cognitive artifact, such as an educational software package, to the learning environment will necessarily add to this complexity. The identification by Norman (1991) of two views of the use of a cognitive artifact - the system view and the personal view- illuminates this notion of increased complexity. From the system view the artifact, in this case the educational software package, enhances cognitive performance in some identifiable ways. In this sense the cognition is shared between the user and the artifact, theoretically making completion of the task by the user easier. From the personal view the artifact changes the nature of the task, and new skills and concepts may need to be acquired. The delegation of some tasks to the artifact and the associated need to learn new skills and concepts makes human-computer interaction a very complex process.

2.5.1 Cognitive complexity in educational software environments

When software is used to support learning, in addition to understanding the concepts and ideas in the area being studied, students also need to learn to manage software effectively. This may involve learning which is essentially irrelevant to the learning task in hand. However, for the effective use of the software to assist learning, students need to learn how to *apply* the software to the current learning task; a high level process which demands both an appreciation of the learning task and how to operate the software. Birnbaum (1990) provides a succinct distinction between the three types of activities associated with the use of information technology (IT) in an educational context: task-intrinsic activities, computer-intrinsic activities, and IT-applicational activities. These types of activities can be thought of as constituting three separate but related domains.

The task-intrinsic domain relates to activities which are directly linked to the learning task. For learning to take place these tasks need to be successfully completed, irrespective of whether or not educational software is used to assist their completion. Within this domain there may be two distinct task areas; (i) an area relating to concepts concerned with the specific topic being studied, and (ii) an area concerned with subsumed and prerequisite concepts related to the general area of study. Successful learning will typically involve students using and relating concepts from both of these task areas.

Tasks in the computer-intrinsic domain are concerned with the efficient and effective management of software. It is clear that students must be able to operate software effectively in order to use it, but the competent use of the features of a software package does not guarantee that learning will take place. Thus the ability of students to complete tasks in this domain is a necessary, but not sufficient, condition for the use of the software to support learning. Two areas can be distinguished in this domain; (i) an area concerned with interaction with the interface, ranging from basic system features such as printing and saving files, through the full use of operating systems such as MS-DOS and Microsoft *Windows*, to the use of application specific features, and (ii) an area concerned with the perception of the system.

The IT-applicational domain relates to activities which are required for the application of the software to complete the learning task in hand. The extent to which the successful completion of these tasks facilitates learning determines how effective the software is in supporting learning. The functions that the software provides must be relevant to concepts related to the learning task. This implies two areas of relevance. The functions obviously need to be relevant to the specific topic being studied, but they may also need to be relevant to subsumed and prerequisite concepts in the general area of study.

The characteristics of the task-intrinsic, computer-intrinsic, and IT-applicational domains are illustrated by Birnbaum (1990) with reference to the use by students of a desk-top publishing package to produce a poster:

> [...] it is important to understand that any IT-related task consists of three types of problem-solving activities: type one is intrinsic to the context and the task (e.g., how best to design a poster and leaflets for the school dance) - we might call this the *task-intrinsic activity*: type two belongs to the software itself (e.g., how do I use this desktop publishing package) - we might call this the *computer-intrinsic activity*; and type three combines the two (e.g., how do I apply this desktop publishing package to the poster design task) - we might call this the *IT-applicational activity*. (Birnbaum, 1990, p. 92)

The discussion of the task-intrinsic, computer-intrinsic, and IT-applicational domains highlights the potentially complex mix of skills and concepts involved in the use of educational software to support learning. Within this environment it is possible to identify three levels of cognitive complexity that are defined in terms of an identification of domains and the relationships between areas in each of these domains. These levels are shown by the "jigsaw" representation shown in in Figure 2.3. Level 1 issues fit with Level 2 issues through the relationships between the two computer intrinsic and the two task intrinsic areas, and Level 2 issues fit with Level 3 issues through the relationship between the two Level 2 issues.

Level 1 tasks are restricted to the task-intrinsic and the computer-intrinsic domains. At this level no deliberate attempt is made to link learning in task areas within the same domain. Thus within the task-intrinsic domain, Level 1 tasks are simply concerned with learning concepts which are either specific to a given topic or which are related to a general area of study, without any attempt to link learning in each of these two task areas. Similarly within the computer-intrinsic domain no attempt is made to consider deliberately the links between system features and the functionality of software applications. This analysis leads to the identification of four evaluation issues at this level as shown in Figure 2.3: (i) concepts specific to a given topic within a general area of study, (ii) subsumed and prerequisite concepts from the general area of study, (iii) system features, and (iv) application features.

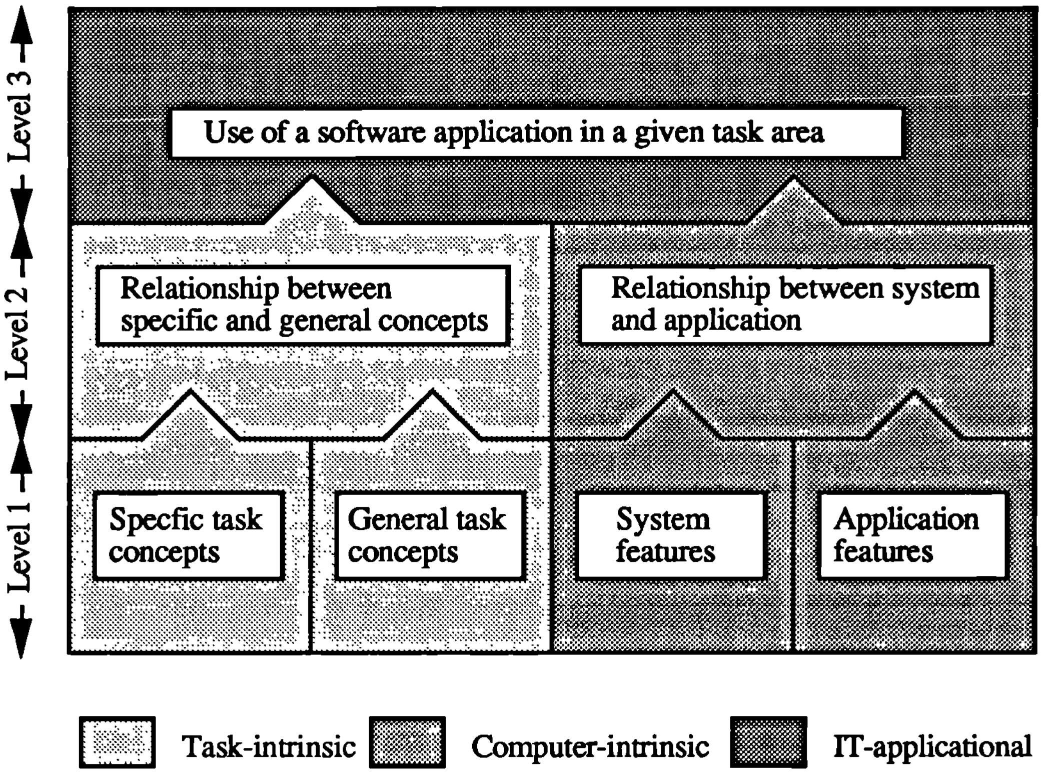


Figure 2.3: Levels of cognitive complexity in educational software environments

Level 2 tasks are also restricted to the task-intrinsic and the computer-intrinsic domains. However, at this level the relationships between task areas in the same domain are considered. In the task-intrinsic domain this involves a consideration of relationships between concepts specific to a given topic and subsumed and prerequisite concepts from the general area of study. In the computer-intrinsic domain this involves a consideration of the relationship between system features and functions of the educational software application. Considering the relationships between the task areas within each of these task domains results in two evaluation issues at this level as shown in Figure 2.3: (i) relationships between specific and general concepts, and (ii) the relationship between the system and application.

Level 3 tasks are only found in the IT-applicational task domain. At this level the use of the application to complete the current learning task is considered. This leads to the critical evaluation issue of how the functions afforded by the software relate to the cognitive demands of the learning task. Considering this issue typically requires the designer to make complex decisions as to how the software will be matched to specific learning requirements. This level subsumes both the inherent complexity of the separate task areas in the task-intrinsic and computer-intrinsic domains (Level 1), and the complexity due to the relationships between task areas in each of these domains (Level 2). Thus if the Level 1 and Level 2 concepts associated with the task-intrinsic and computer-intrinsic domains are easy to grasp it is likely that the application of the software to facilitate learning will be relatively straightforward. In contrast, if Level 1 and Level 2 concepts in both of these domains are hard to grasp, it is likely that the effective application of the software to support learning will be problematical. If the Level 1 and Level 2 concepts associated with the task-intrinsic domain are relatively simple it is debatable whether it is necessary to use educational software to support learning. If, in addition, concepts at these levels in the computer-intrinsic domain are difficult, it implies that the use of educational software is inappropriate. In contrast, the reverse situation, with complex Level 1 and Level 2 concepts within the task-intrinsic domain and relatively simple Level 1 and Level 2 concepts within the computer-intrinsic domain, implies that the use of educational software is beneficial. In this case easily understandable computer-based activities are available to tackle difficult learning problems.

2.5.2 Designer and learner models of software complexity

diSessa (1986) identifies three types of model that are employed by users to understand the way software works, and by implication the way that the system operates. The first type of model he termed a structural model, which he noted as being identical to the idea of a surrogate model introduced by Young (1981, 1983). Such models "conjure up the image of a sort of replacement machine located in the mind on which one can run experiments and envision results without touching the actual machine" (diSessa, 1986, p. 202). This idea is illustrated by Young (1981) in his description of the implied register models held by users of calculators:

> Implied register models [...] are, roughly, "cover stories". i.e. simple descriptions of hypothetical machines whose behaviour matches that of the calculators themselves. [...] these models provide an account of how the calculator *works* rather than saying directly how it can be *used* to achieve a desired result. (Young, 1981, p.52)

Thus a surrogate model mimics the structure of the system at a level which is understandable to the user. Of course, a surrogate model does not provide a complete description of the structure of a system. However, a good surrogate model must provide a model which is both comprehensive and accurate at the level at which the user perceives the system. The choice of the level at which to pitch a surrogate model is critical. If the level is too low, that is too close to the core language, the user perceived difficulty will be too high and the model will be too difficult for the user. If the level is pitched too high, the description offered by the model will be too superficial to be of genuine use to the user.

The second type of model diSessa terms a functional model, that is a model which is only concerned with how to achieve results without a concern for how these results are effected by the operation of the system. He describes such models in the following terms:

> The general schematic of functional models is that one has a descriptive frame [...] that includes recognisable objects and actions [...]. Then the user can understand computational constructs and actions as they function in this frame. Functional models might be described simply as rules, [...] but to think of them in this way ignores the fact that such a rule is memorable precisely because it fits into the previously understood scheme of goals and means [...]. (diSessa, 1996, p. 208)

Young (1981) gives the example of an algebraic calculator, that is one that permits entry in an algebraic form using parentheses, as a classic functional model. He makes the point that it is very difficult to develop structural models of such calculators and that users inevitably have recourse to the use of functional models.

A functional model only provides a model for the task in hand, and a different functional model may be required for a different task. Thus users typically hold a number of functional models. This contrasts with a structural model which is designed to provide a comprehensive representation.

The third type of model is described by diSessa as a distributed model. Such a model is described by diSessa as follows:

> I refer to models accumulated from multiple, partial explanations as *distributed models*. [...] These models represent a patchwork collection of pre-existing ideas in the learner, "corrupted to new ends. They don't appear nearly as "model-like" as structural or functional models. But if "model" is too strong a word to describe them "understanding" is not. (diSessa, 1986, p. 209)

Both the designer and the learner will hold models of the system and how it can be manipulated to complete the task in hand. Norman (1983) acknowledged the existence of the designer's model as a conceptual model of the system, and the learner's model as the user's mental model of the target system. He also introduced the notion of a designer's model of the learner's model:

> [...] we need a conceptual model of the system: call the conceptual model of $t, C(t)$. And now let the the user's mental model of that target system be called $M(t)$. We must distinguish between our conceptualisation of a mental model, $C(M(t))$, and the actual mental model that we think a person might have, $M(t)$. (Norman, 1983, p.11)

In terms of diSessa's analysis the learner's model will typically be distributed, incorporating a repertoire of functional models, with the possible addition of a surrogate model of the structure of the system. The designer's model will also typically include a set of functional models, but it will also certainly incorporate a model of the system at some level. Some designers (for example, those intimately connected with the coding of a software application) may have a number of structural models, ranging from a representation of the system in the core language through to surrogate representations at different levels. Other designers, may simply possess a surrogate model of the system. The designer's model of the learner's model needs to take account of the differences between both structural and functional learner and designer models Notably it is important to establish what, if any, surrogate model the learner holds, and the relationship of this model to the structural models held by the designer.

Norman's analysis leads to the idea of three versions of the jigsaw representation of software complexity; a user's model, a designer's model, and a designer's model of the user's model. In this section each of the seven design issues identified in this Section 2.6.2 will be considered in terms of each version of the model, leading to a framework which will provide the basis for the explanatory component of an evaluation model. A summary of the framework is provided in Table 2.1

2.5.2.1 Evaluation issue 1: concepts associated with a specific topic

A designer will use accepted ideas and concepts about a specific topic in the design of an application. For example, in developing a simulation of a nuclear reactor a designer would develop a mathematical model based on the laws of physics and established engineering practice. It is not always the case that users of educational software will share the same concepts and ideas; in fact the norm is for learners to be unaware of critical ideas and concepts in the topic area and not to fully understand concepts they are aware of. The potential conflict between a designer's model, based on accepted ideas and concepts, and a learner's model based on misunderstood or incomplete concepts needs to addressed through the designer's model of the learner's model.

Designer's model

- What accepted specific and general concepts about a given topic is the design of the software based on?

Learner's model

- What are the typical misunderstandings and gaps in knowledge in this area?

Table 2.1: Evaluation issues

Level	Issue	Designer's model	Learner's model	Designer's model of the learner's model
1	1	Accepted concepts in a given topic	Misunderstandings or incomplete knowledge in a given topic	Approach to dealing with misunderstandings and incomplete knowledge in a given topic
1	2	Accepted prerequisite concepts and skills	Awareness of prerequisite concepts and skills	Significance of prerequisite concepts and skills
1	3	Representation of the system that is used in the design	Understanding of the state of the system	Approach to representing the state of the system to the learner
1	4	Definition of application functionality	Perception of application functionality	Sub-set of application functionality required to complete a given task
2	5	Assumed links between specific and general concepts	Perceived links between specific and general concepts	Recognition of weak or non-existent links between specific and general concepts
2	6	Definition of the yoking between system space and task space	Perceived links between the system and the task	Awareness of mismatches between defined and perceived links between the system and task
3	7	Intended application of the software to support learning	Actual application of the software to support learning	Mismatches between intended and actual use

Designer's model of the learner's model

- Does the design of the software take account of learners' typical misunderstandings and gaps in knowledge in this area?
- Is there any attempt to overtly represent learners' misunderstandings about a given topic?

2.5.2.2 Evaluation issue 2: subsumed and prerequisite concepts

Understanding a given topic typically involves understanding other concepts within the general area of study and using generally applicable problem solving and process skills. As with concepts specific to the area of study, learners will often misunderstand these general concepts while the designer will adopt the accepted view of these concepts. The designer will also usually take for granted that learners possess appropriate problem solving and process skills.

Designer's model

- What prerequisite concepts are assumed to be understood by learners?
- What problem solving approaches does the design of the software support?
- What process skills does the design of the software assume that learners possess?

Learner's model

- Are learners aware of prerequisite concepts and do they understand them?
- Do learners employ idiosyncratic problem solving techniques when using the software?
- Do learners possess the assumed process skills?

Designer's model of the learner's model

- Are attempts made to highlight the importance of prerequisite concepts?
- Is an attempt made to provide remedial assistance to cope with problems in understanding prerequisite and subsumed concepts?
- Does the design of the software take account of learners using idiosyncratic problem solving approaches?
- Is allowance made for learners not having a full range of appropriate process skills?

2.5.2.3 Evaluation issue 3: system features

In implementing the design of a software package the designer will define the relationship between the system and the application, that is the relationship of the core language with the input and output languages. This definition will control the effects of operations executed while the application is being used. For example, diSessa (1985, 1991) discusses the merits of adopting dynamic as opposed to lexical scoping of variables in the design of *Boxer*. Typically, the user is expected to be either unaware of the relationship between system and application, or cognisant of the relationship in a contrived or limited fashion.

Designer's model

- What structural model of the system did the designer use when developing the application?

Learner's model

- What surrogate models of the system do learners hold and at what levels are they?

Designer's model of the learner's model

- Are users expected to have a structural model of the system, and, if so, in what form are they expected to hold this model?
- Is any attempt made to represent a surrogate system model to the user?

2.5.2.4 Evaluation issue 4: the functionality of educational software

The designer will decide on the range of operations by defining the input language and the range of representations by defining the output language. These decisions define the functionality of an application. Often there is a sub-set of essential operations which define the core functionality of the software. For example, the basic editing operations provide the core functionality in a word processing package, while more elaborate formatting commands extend this core functionality. Thus the functional models implied by the design of a system can be regarded as core and optional.

Designer's model

- What core and optional functional models is the design of the system predicated on?

Learner's model

- What functional models of the system do learners typically hold?
- How comprehensive is the set of functional models that a learner holds? Does the learner understand the full set of core functional models? What functional models in addition to the core set are they aware of?

Designer's model of the learner's model

- What are the differences in terms of extent and character between learner and designer functional models?

2.5.2.5 Evaluation issue 5: the relationship between specific and general concepts

The designer's model will typically assume that learners are capable of seeing the links between topic specific and general aspects of learning. However, the learner's model may not take account of these links. In some cases a learner's appreciation of the links between specific and general concepts may be tacit. The designer's model of the learner's model needs to take account of the possibility of weak, and possibly non-existent, links between learner's understanding of topic specific and general concepts.

Designer's model

- What links between specific and general concepts are assumed in the software design?

Learner's model

- Are learners aware of links between topic specific and general concepts, and do they understand the significance of these links?

Designer's model of the learner's model

- Is any account taken of the possibility of weak, and possibly non-existent, links between a learner's understanding of topic specific and general concepts?

2.5.2.6 Evaluation issue 6: the relationship between system features and application functionality

The designer's model is based on a knowledge of the relationship between the system structure and the software functionality. Learner's will not in most cases have this knowledge. Usually they will be unaware of a relationship. However, in some cases the designer will incorporate a surrogate model of the system, which will typically be at a more superficial level or in a different form than the defining model.

Designer's model

- What decisions have been made in defining the relationship between the system and the interface?

- At what level is it possible to represent the relationship between the system and the interface?

Learner's model

- Are learners aware of a relationship between the system and the interface?
- If learners are aware of a relationship between system and application what surrogate model do they employ? What is the semantic distance between these representations?
- Does a lack of understanding, or a partial understanding, of the relationship between system and application cause problems for learners?

Designer's model of the learner's model

- Is an attempt made to use a surrogate model to represent the relationship between system and application, that is, to minimise the semantic distance?
- If a surrogate model is used how does this representation correspond to the basic defining relationship?

2.5.2.7 Evaluation issue 7: the use of a software application for a specific learning task

The designer's model is based on the assumption that the representations and functions in a software application are a legitimate and useful representation of the specific and general concepts associated with a given topic. The learner's model is defined in terms of those features of the application which they find useful in learning the topic in question. The designer's model of the learner's model needs to assess whether the functions and representations are meaningful to learners , that is, to assess the user perceived difficulty.

Designer's model

- In what ways is it envisaged that the software application will be used?
- Does the design of the software application assume that learners are aware of, and understand, the full range of specific and general concepts?
- If it is not assumed that learners understand a full range of specific and general concepts, does the software design specifically address this lack of knowledge?

Learner's model

- How close is the match between the designer's anticipated use of the software application and the actual use by learners? More specifically do the learners use the assumed functions and interpret the representations in the expected way?
- Does the design make learners aware of specific and general concepts and skills?

- What assistance is provided to learners to understand specific and general concepts?

Designer's model of the learner's model

- Are unexpected uses or unforseen interpretations of representations catered for?
- Do the functions and representations in the software highlight the significance of relevant concepts and provide assistance in mastering these concepts?

2.5.3 Combining explanation and prediction

As shown in Figure 2.4, the complexity framework developed in Section 2.5.2 provides a basis for combining an explanatory component, based on an inner psychological theory and a consideration of the system, with a prescriptive component provided by the GOMS model, to provide a comprehensive evaluation framework for educational software.

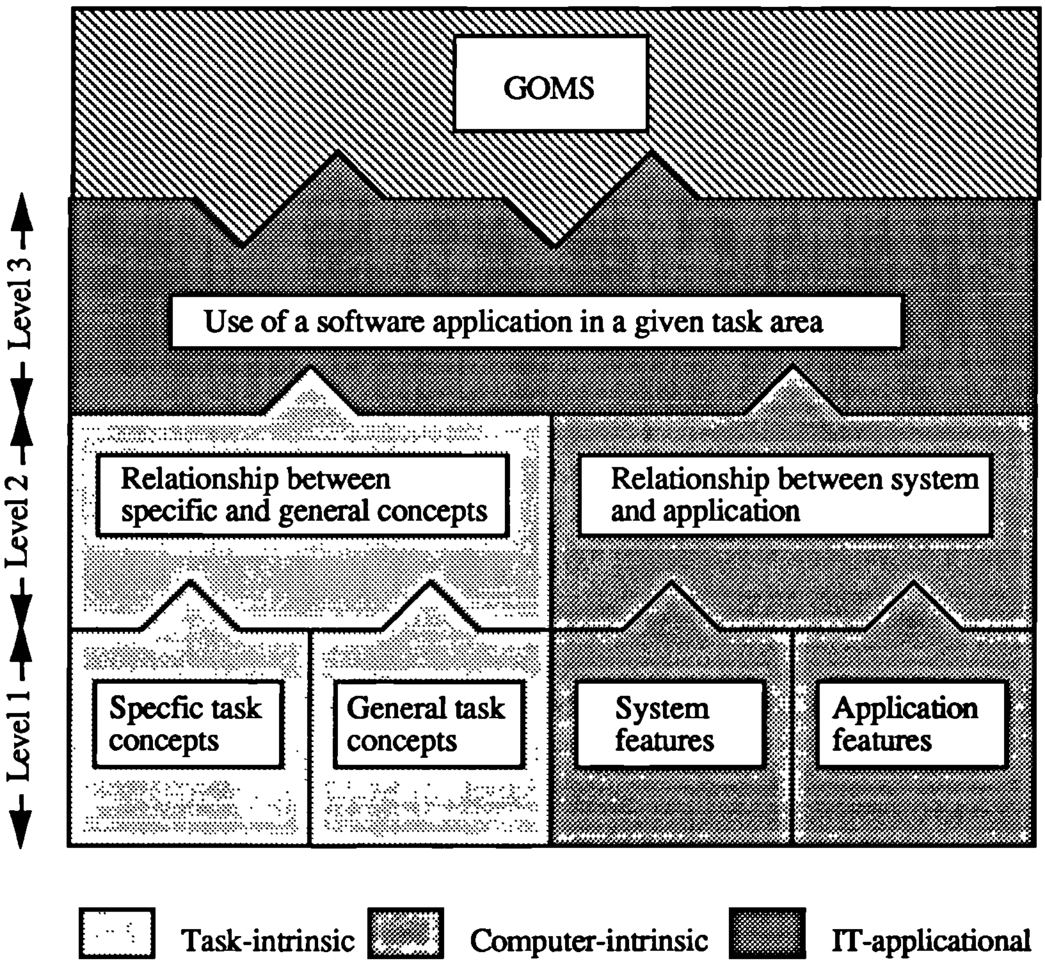


Figure 2.4: The jigsaw software complexity framework applied to the evaluation of educational software.

The predictive and explanatory components of the model interact as shown in Figure 2.4; GOMS identifies design problems and the explanatory component provides an explanation. Although the identification of the design problem will be at Level 3, the interlocking nature of the levels enables an explanation of the problem to be made at Level 3, 2 or 1.

The three versions of the application of the jigsaw complexity framework introduced by a consideration of the designer's model, the learner's model, and the designer's model of the learner's model allow the development of a comprehensive model, as shown in Figure 2.5.

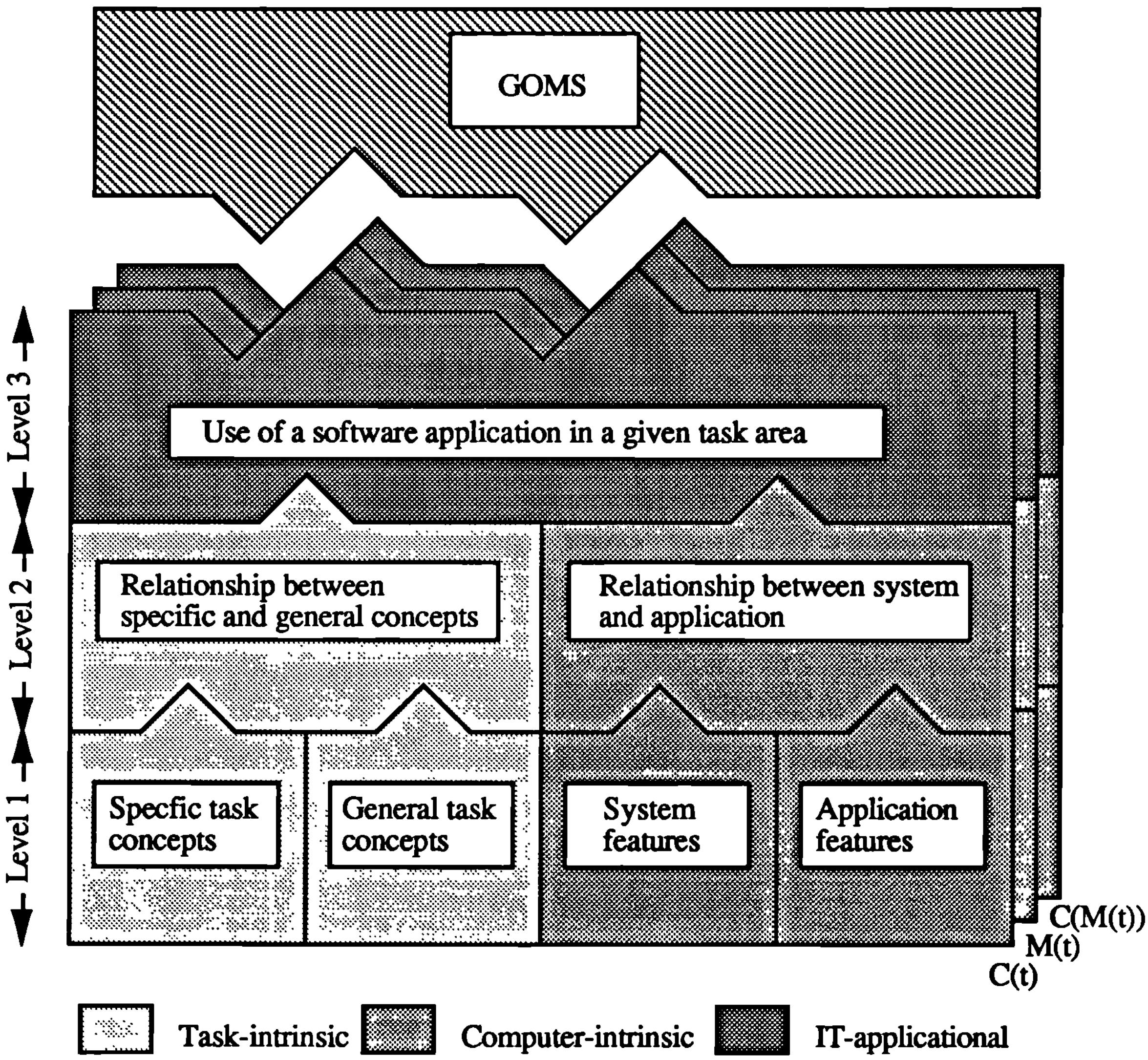


Figure 2.5: The Jigsaw Model for the evaluation of educational software

In essence this development indicates that the explanation of a design problem identified by the predictive GOMS component of the model should be made from three perspectives in order to achieve a truly comprehensive evaluation. Thus the resulting Jigsaw Model aims to provide a comprehensive predictive and explanatory account of the relationship between learners' and the designer's mental models. In this sense it provides an in-depth consideration of what Squires & McDougall (1994) term the interaction between the perspectives of the student and the designer.

Squires & McDougall also describe two other interactions which are relevant to the evaluation of educational software; the interaction between the perspectives of the teacher and the student, and the interaction between the teacher and the designer. The former of these two interactions is primarily concerned with effect that the use of educational software has on the pedagogic and social environment of the classroom, and latter is primarily concerned with the relevance of educational software to the curriculum. The Jigsaw Model does not address either of these interactions.

2.6 Summary

Existing evaluation frameworks for educational software do not take due account of human-computer interaction issues. This omission is becoming more important as the sophistication of the software typically used in schools increases. In particular the widespread use of direct manipulation software in education needs to be considered. Established usable models for human-computer interaction are inadequate for representing interaction in a direct manipulation context. The GOMS model has been identified on the basis of four criteria - possibility of extension, application in diverse task areas, simplicity, and compatibility with the target task domain - as a suitable model to act as basis for the development of a comprehensive evaluation framework for the cognition associated with the use of direct manipulation educational software. Such a model has been developed with two components; a predictive component provided by the application of the GOMS approach, and an explanatory component based on a consideration of software complexity.

Chapter 3

An educational task domain: limiting factors in photosynthesis

Bioview, the direct manipulation software tool which is the focus of this research, provided an environment to explore the relationships between interacting variables. Limiting factors in photosynthesis, an example of interacting variables which is featured in A-level biology syllabuses, provided a task domain for the use of *Bioview*.

Photosynthesis in general is regarded as a difficult topic (Simpson & Arnold, 1982a; Johnstone & Mahmoud, 1980; Finley, Stewart & Yarroch, 1982). The research into students understanding of photosynthesis indicates that students experience three areas of difficulty; (i) the concepts which are used to explain photosynthesis are in themselves complex, giving rise to diverse and fundamental learning difficulties, (ii) photosynthesis concepts require a prior grasp of more basic concepts, such as food, energy, energy conversions, living things, chemical change, and cells, which students also find difficult (Simpson & Arnold, 1982b; Stavy, Eisen & Yaakobi, 1987), and (iii) photosynthesis is a very broad area of study involving concepts from diverse areas, with the relationships between these areas typically poorly understood by students (Waheed & Lucas 1992). These difficulties lead to common misconceptions which are extensively documented in the research literature (see Bell (1985) and Wood-Robinson (1991) for reviews of this literature). These misconceptions are briefly described in Section 3.1.

Students' understanding of interacting variables is considered in Section 3.2, and the specific research into students understanding of limiting factors in photosynthesis is discussed in detail in Section 3.3.3.

3.1 Students' understanding of photosynthesis

Research into students' understanding of photosynthesis is extensive, with a considerable amount of data now collected which allows the formation of a comprehensive view of misconceptions in this area. The classification of concepts associated with photosynthesis into ecological, biochemical, anatomical-physiological, and energy change concepts (Waheed & Lucas, 1992) is adopted as a framework for briefly reviewing the findings of this research.

3.1.1 Ecological concepts

An understanding of photosynthesis from an ecological perspective involves an appreciation of the role of photosynthesis as a mechanism for enabling the transfer of energy from the sun to plant and animal life, the recycling of carbon dioxide and oxygen, the role of soil, and the role of light.

A study reported by Eisen & Stavy (1988) found a marked difference between the understanding of autotrophic feeding by biology majors and biology non-majors; 90% of a sample of biology majors mentioned the sun's role in autotrophic feeding, while the non-biology majors did not mention the sun's role in the carbon cycle or the food chain. Stavy et al. (1987) found that students had incomplete and superficial conceptions of the sun's role in the food chain. Soyibo (1983) reports the widespread belief that oxygen passes into plant leaves and carbon dioxide passes out. Wandersee (1983) in a large cross-age study (1405 students in the age range 10 to 18) of students' understanding of photosynthesis found that students believe that during photosynthesis plants give off mainly carbon dioxide, water vapour moves into the leaf, and plants take oxygen out of the air. The studies by Eisen & Stavy (1988) and Stavy et al. (1987) report that students do not appreciate the role of plants in recycling oxygen. A large number of studies report that significant numbers of students consider soil to be the source of food for plants (Arnold & Simpson, 1980; Simpson & Arnold, 1982a; Roth, Smith & Anderson, 1983; Wandersee 1983; Bell & Brook, 1984; Smith & Anderson, 1984; Driver, Child, Gott, Head, Johnson, Worsley, & Wylie, 1984; Bell 1985; Stavy et al., 1987). The role of light in photosynthesis has been considered by a number of studies (Roth et al., 1983; Wandersee, 1983; Bell & Brook, 1984; Smith & Anderson, 1984; Stavy et al., 1987; Eisen and Stavy, 1988). The results indicate that students are aware at a superficial level of the importance of light to plant growth.

3.1.2 Biochemical aspects of photosynthesis

Understanding the biochemical aspect of photosynthesis involves an appreciation of (i) the role of environmental factors such as temperature, light intensity, carbon dioxide concentration, and water, (ii) the role of chlorophyll, and (iii) the way in which various factors can limit the rate of photosynthesis.

Students generally consider light to be important without fully understanding the role that light plays in the chemistry of photosynthesis. This lack of understanding is evident in the results of research conducted by Amir & Tamir (1993) into the effects of the use of the terms "light" and "dark" reaction for respectively the light dependent and the light independent chemical reactions in photosynthesis. They identified a common misconception that the "dark" reaction is temporally separated from the "light" reaction and occurs only at night or in the dark. Many students do not know that carbon dioxide from the atmosphere is the source of carbon which is reduced by plants during photosynthesis to produce sugar which is the plant's food. Given this fact, it is not surprising that there are widespread misconceptions as to what constitutes a plant's food (see for example Simpson & Arnold 1982a, 1982b; Roth et al. 1983; Soyibo 1983; Wandersee 1983; Bell & Brook 1984; Driver et al. 1984; Smith & Anderson 1984; Eisen & Stavy 1987, Stavy et al. 1988; and Barker & Carr 1989). In general few students appear to understand the photochemical origin of plant food, preferring an "everyday" view of the origin of food as materials which are eaten, rather than the biologist's view of food as the material which is used in respiration. Misconceptions about the role of chlorophyll are common (Simpson & Arnold 1982a, Wandersee 1983, and Soyibo 1983). Wandersee (1983) found that these misconceptions were amongst the most tenacious, with students' understanding in this area showing the least improvement with time compared with most other concepts.

3.1.3 Anatomical-physiological concepts

Understanding the anatomical aspects of photosynthesis involves a consideration of the significance of leaf structure on photosynthesis, and understanding of the physiological aspects involves a consideration of respiration in plants.

The results of research into students understanding of respiration in plants (Simpson & Arnold, 1982a; Wandersee, 1983; Stavy et al., 1987; Eisen & Stavy, 1988; Haslam & Treagust, 1987; Amir & Tamir, 1990) reveals that there is widespread confusion as to whether plants respire or not and about nature of respiration. There also appears to be widespread confusion between the processes of respiration and photosynthesis, with associated misconceptions about the time at which respiration occurs.

Limited research on leaf structure and students' understanding of photosynthesis is reported in the literature. In their study of how students aged 13-15 understand photosynthesis, Stavy et al. (1987) deliberately decided not to deal with students' understanding of plant structure because "its concrete nature makes the subject easy to grasp" (Stavy et al., p. 106). However, Wandersee (1983) did detect some misconceptions that students hold about the relationship of leaf structure to photosynthesis, foe example the leaf's main job is to capture the rain and water vapour in the air.

3.1.4 Energy conversions

A full understanding of photosynthesis requires an understanding of the concept of energy and how energy can change from one form to another. Given the difficulties that students experience in understanding energy (see Brook (1985) for a review of the literature) it would seem reasonable to assume that this aspect of photosynthesis would present significant learning difficulties.

The results of the study by Simpson & Arnold (1982b) confirmed that problems with understanding energy can hinder an understanding of photosynthesis. They found that primary school students were happy to use the term energy, and that secondary school students could name different forms of energy. However quite poor performances were found in the classification of forms of energy. The findings of a number of studies confirm that students experience difficulties in understanding the energy conversions that take place during photosynthesis. Wandersee (1983) discovered a belief that plants convert energy from the sun directly into matter. Haslam & Treagust (1987) found that a high proportion of students did not comprehend photosynthesis as an energy conversion process, viewing it as an energy providing process. In the study conducted by Eisen & Stavy (1988) a question related directly to harnessing the sun's energy was posed - "Describe the energy transfer in the photosynthetic process". In answer to this question only 2.5% of non-biology majors described the energy transfer process in photosynthesis, and 68% of biology majors provided a correct explanation of the process.

3.2 Interaction between variables

Students are typically taught to conduct investigations by adopting a simplistic strategy which has been described as "change one thing at a time and keep all else constant" (Lucas & Tobin 1987) or "isolation of variables" (Dawson and Rowell 1986). In this strategy instances of the state of the system or environment being investigated are defined that only differ in the value of one variable. The behaviour of the two instances is then compared. The *interaction* between variables is not considered.

It is possible to criticise the isolation of variables strategy on two major grounds. Firstly, it is not possible to guarantee that all the variables are kept constant. Lucas & Tobin (1987) illustrated this point with respect to an experiment to determine how the diameter of a tube affects its strength when it is used as a girder and as a column. If the isolation of variables strategy is adopted all variables, such as the length of the tube, should be kept constant as the diameter is varied. How should the thickness of the tube wall be kept constant? Should the thickness be kept at the same absolute value or the same value to relative to the diameter? Again, if the mass is kept constant how should the thickness be varied? The common investigation of the variables that control the period of a pendulum illustrates this point further. It is conventionally assumed that the variables of interest are the length of the pendulum and the mass of the bob, but should the angle of the oscillation and the mass of the string be considered as well? As Lucas & Tobin point out, experimentally it is as justifiable to conclude that it is the mass of the string, rather than the length of the string, that affects the period of the pendulum. They conclude that it is impossible to control all variables, and that in reality it is only possible to control variables which are perceived as relevant; a perception which will depend on the general ideas that the experimenter has of the phenomenon being investigated. This leads Lucas & Tobin to suggest a revised statement of the isolation of variables strategy:

> We cannot keep all else constant, but it might be possible to use essentially similar procedures if we modify our paradigm and say that what we really mean is "that if we can think of a number of variables that might affect the dependent variable, then we should keep all but one of these relevant variables constant while we systematically vary the remaining one. (Lucas & Tobin, 1987, p. 686)

Thus experimental investigations necessarily involve the identification of principal variables which are perceived as relevant to the experiment in hand.

Secondly, a failure to consider the interaction between variables leads students to consider simply the effect of one variable rather than investigate the *optimum* conditions. Shayer (1986) observed that although students at Piaget's concrete stage may have some appreciation of how to make a "fair test", that is control variables, they tend to believe that the best results from an experiment can be obtained by varying all the variables about which they have a preconception. This leads to a probable confounding of variables in an investigation. At the formal stage, he states that students understand that randomly changing variables will lead to ambiguous results, but that they may commit a more subtle, but equally significant, error by controlling everything in sight - an approach which can lead to misleading or incorrect conclusions. Lucas & Tobin (1987) provide two examples to illustrate this point - the investigation of the digestion of starch by saliva, and lung damage caused by polluted air. Variables that are typically considered as relevant to an investigation of the digestion of starch include temperature, pH, substrate concentration, and amount of enzyme. How should the pH be kept constant? For example, should it be kept at a neutral value of 7, a high acid value, or the pH of the mouth? This is a critical decision as the effect observed for temperature depends on the pH value. In a similar fashion they discuss how there is an interaction between the effect of the size of dust particles in the atmosphere and the concentration of sulphur dioxide which leads to lung damage being observed at lower concentration values that would be predicted by investigating the effects of these two variables independently. They also discuss an investigation into the conditions required for rusting to take place - an investigation which is a common feature of most elementary chemistry courses - as an example the interaction of variables

The essential problem with the isolation of variables strategy is that it does not recognise the interaction between variables, as stated by Fisher (1960):

In expositions of the scientific use of experimentation it is frequent to find an excessive stress laid on the importance of *varying the essential conditions only one at a time* [Fisher's emphasis]. The experimenter interested in the causes which contribute to a certain effect is supposed to, by a process of abstraction, to isolate those causes into a number of elementary ingredients, or factors, and it is often supposed [...] that to establish controlled conditions in which all of these factors except one can be held constant, and then to study the effects of this single factor, is the essentially scientific approach to an experimental investigation. This ideal doctrine seems to be more nearly related to expositions of elementary physical theory than to laboratory practice in any branch of research.

[I]n the state of knowledge or ignorance in which genuine research, intended to advance knowledge, has to be carried out, this simple formula is not very helpful [...] We have usually no knowledge that any one factor will exert its effects independently of all others that can be varied, or that its effects are particularly simply related to variations in these other factors [...] If the investigator [...] confines his attention to any single factor, we may infer that he is the unfortunate victim of a doctrinaire theory as to how experimentation should proceed, or that the time, material or equipment at his disposal is too limited to allow him to give attention to more than one narrow aspect of the problem" (Fisher, 1960, pp. 93-94)

Factors which affect the rate of photosynthesis in green plants provide a graphic illustration of the interaction between variables. As in all situations decisions need to be made as to what the principal variables should be. In this case they are usually considered to be temperature, light intensity, and the level of carbon dioxide (see Section 3.3.1 for a discussion of this issue). These three variables interact to determine the optimum conditions for photosynthesis. In particular, at the optimum temperature it is possible for one of both the light intensity and the level of carbon dioxide to limit the rate of photosynthesis. Students experience significant problems in understanding the concept of limiting factors in photosynthesis (Amir & Tamir 1989, 1990), making this an appropriate topic for evaluating the educational use of *Bioview*.

In Sections 3.3.1 and 3.3.2 the concept of limiting factors in photosynthesis is considered in detail, and the findings of research into students' understanding in this area are discussed in detail in Section 3.3.3.

3.3 Limiting factors in photosynthesis

The concept of a "limiting factor" is important in understanding a range of biological processes. This concept is dealt with almost exclusively in the school curriculum in connection with photosynthesis, as reflected in the way this topic features in sections on photosynthesis in school textbooks (see for example, Roberts, 1981; Nuffield Advanced Science, 1985; Phillips & Chilton, 1989; Green, Stout & Taylor, 1990; Toole & Toole, 1991). However, students appear to have some problems with understanding the concept in this context, as highlighted by Amir & Tamir (1989):

> Despite the intensive treatment of the concept in textbooks and the classroom, students' achievements in test items dealing with the concept has been found to be quite low. In the 1986 Matriculation examination, taken by 5000 Israeli students, the average achievement on an item dealing with the concept was 66 per cent and was the lowest of 6 items of similar level of difficulty. (Amir & Tamir, 1989, p. 129)

3.3.1 Principal limiting factors

Typically the concept of limiting factors is considered with respect to the effects on the rate of photosynthesis of light intensity, temperature, and carbon dioxide concentration. For example Nuffield Advanced Biology (1985) presents the topic by considering changes in the rate of photosynthesis at different temperatures, whilst Roberts (1981) deals with the topic by considering the changes in the rate with light intensity at different concentrations of carbon dioxide. However, light intensity, carbon dioxide concentration, and temperature are not the only possible limiting factors. Heath (1969) describes the variety of factors which can affect the rate of change of photosynthesis:

> The rate of photosynthesis is controlled by a large number of (i) external and (ii) internal factors, of which the more important are: (i) light intensity (and quality), temperature, carbon dioxide concentration, wind velocity, water supply, nutrient supply; (ii) age, chlorophyll content, enzyme factors, leaf water content, leaf structure, stomatal aperture. (Heath, 1969, p. 113)

In this context Jungwirth (1988) criticises the discussion by Wandersee (1985) of one of the items in the photosynthesis concept test developed by Wandersee (1983). Jungwirth quotes this item as:

> If you were asked where most of the food of plants comes from, which of the following choices would you consider the *best* answer? Mark one.
> (a) the food of plants comes from water
> (b) the food of plants comes from carbon dioxide
> (c) the food of plants comes from the soil
> (d) the food of plants comes from water and air
> (Jungwirth, 1988, p. 161)

In an analysis of students responses to this question Wandersee (1985) states:

> The fourth option was chose by a sizeable number of students who apparently believe both water and air (carbon dioxide) are equally important sources of raw materials plants use to make food molecules. The percentage for this choice increased as grade levels did. (Wandersee, 1985, p. 592)

Jungwirth does not find this surprising, pointing out that both water and carbon dioxide are limiting factors (even though water is only required in small quantities). He criticises Wandersee for confusing quantity with importance. Wandersee replies to Jungwirth by asserting that there are "principal" limiting factors which are more important to consider (particularly from an educational point of view) than others. Whilst acknowledging the importance of "The Law of Limiting Factors" he asserts:

> Yet to teach a student that everything is important is to teach him or her nothing. It is educationally defensible to select principal limiting factors for teaching and testing. (Wandersee, 1988, p. 163)

Wandersee's view is supported by the typical approach taken to experimental design in this area, with studies typically assuming that materials such as water and minerals are in adequate supply, and that the principal limiting factors are light intensity, carbon dioxide concentration and temperature (see Rabinowitch 1951, 1956) The experimental results contained in the PSYNTH datacube (see Section 4.1) are representative of this approach; water and mineral salts are assumed to be in abundant supply with light intensity, carbon dioxide concentration and temperature operating as the only limiting factors.

3.3.2 Representations of the effects of limiting factors

Most textbooks represent the dependence of the rate of photosynthesis on environmental factors graphically as shown in Figure 3.1. The variation of the rate of photosynthesis with light intensity is shown as a family of curves, with each curve corresponding to a fixed value of another limiting factor; either carbon dioxide concentration (see for example Roberts, 1981; Green et al., 1990; Toole & Toole, 1991) or temperature (see for example Nuffield Advanced Science, 1985; Phillips & Chilton, 1989). Each curve consists of three sections: (i) an initial linear part, common to the family of curves, which shows the rate of photosynthesis increasing with light intensity (represented by AB), (ii) an intermediate non-linear section in which the rate of photosynthesis does not increase as rapidly (represented as BC), and a final section in which the rate of photosynthesis remains constant at a maximum value as light intensity is increased (represented as CD).

The shape of these curves corresponds to a common stereotype for graphs representing the variation of rate of photosynthesis with environmental factors. Rabinowitch (1951) characterised these as "Blackman" type curves, after Blackman who postulated a "principle of limiting factors" in connection with photosynthesis. Heath (1969) cites this principle as:

> When a process is conditioned as to its rapidity by a number of separate factors, the rate of the process is limited by the pace of the slowest factor. (Heath, 1969, p. 116)

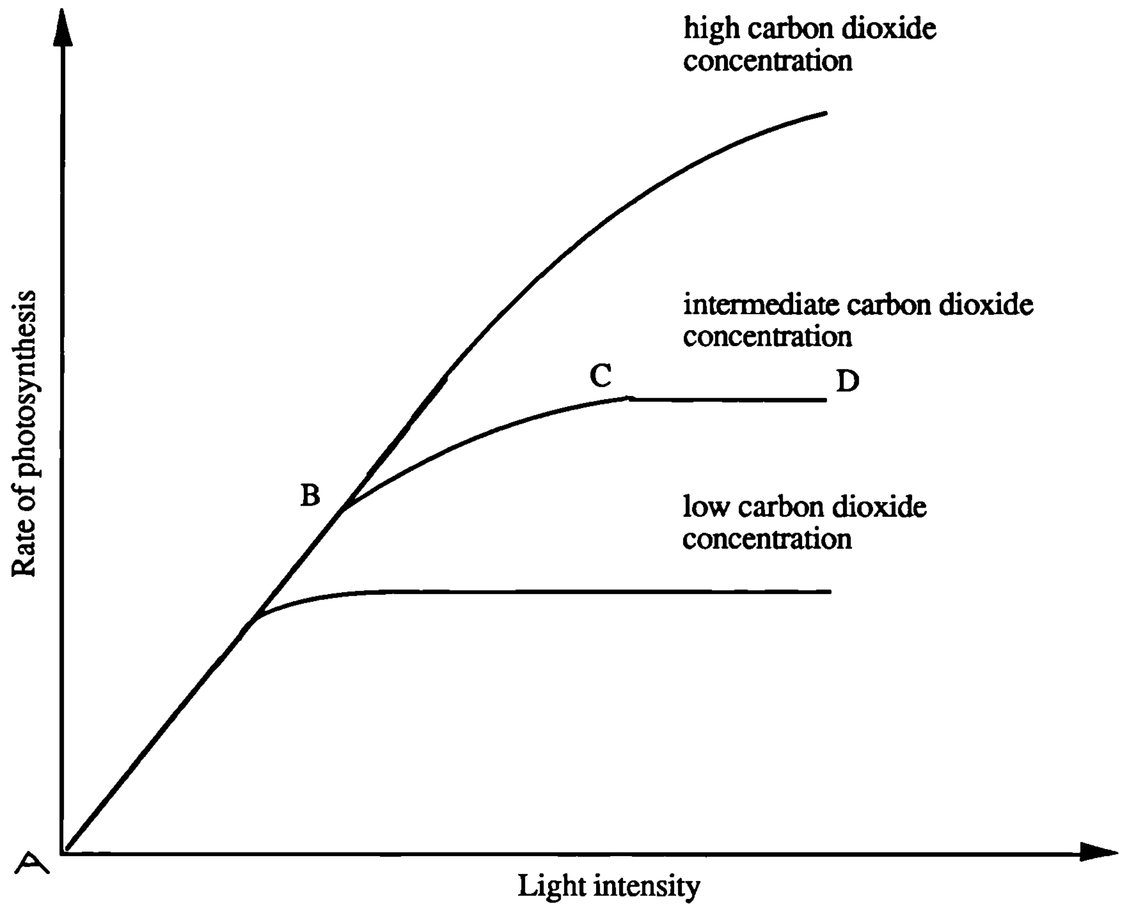


Figure 3.1: Typical graphical representation shown in text books of the variation of the rate of photosynthesis with environmental factors.

This implies that when the rate of photosynthesis is subject to the effects of a limiting factor, only an increase in the value of this factor (and no other potentially limiting factors) will result in an increase in the rate of photosynthesis. A strict application of Blackman's principle indicates a family of rate curves of the type shown in Figure 3.2, in which the rate of photosynthesis at a constant temperature is plotted against light intensity for different fixed values of carbon dioxide.

The curves shown in Figure 3.2 only consist of two distinct sections, in contrast to the three sections typically identified in experimental work and shown in school textbooks. The first section consists of a common linear slope with light intensity as the limiting factor (represented by AB). For light intensities corresponding to this section an increase in intensity results in an increase in the rate of photosynthesis, while an increase in the concentration of carbon dioxide does not affect the rate. The second section consists of a straight horizontal line with carbon dioxide as the limiting factor (represented by BC). For light intensities corresponding to this section an increase in light intensity does not result in an increase in the rate of photosynthesis above a maximum value, while an increase in carbon dioxide concentration results in an elevation of the maximum value of the rate of photosynthesis.

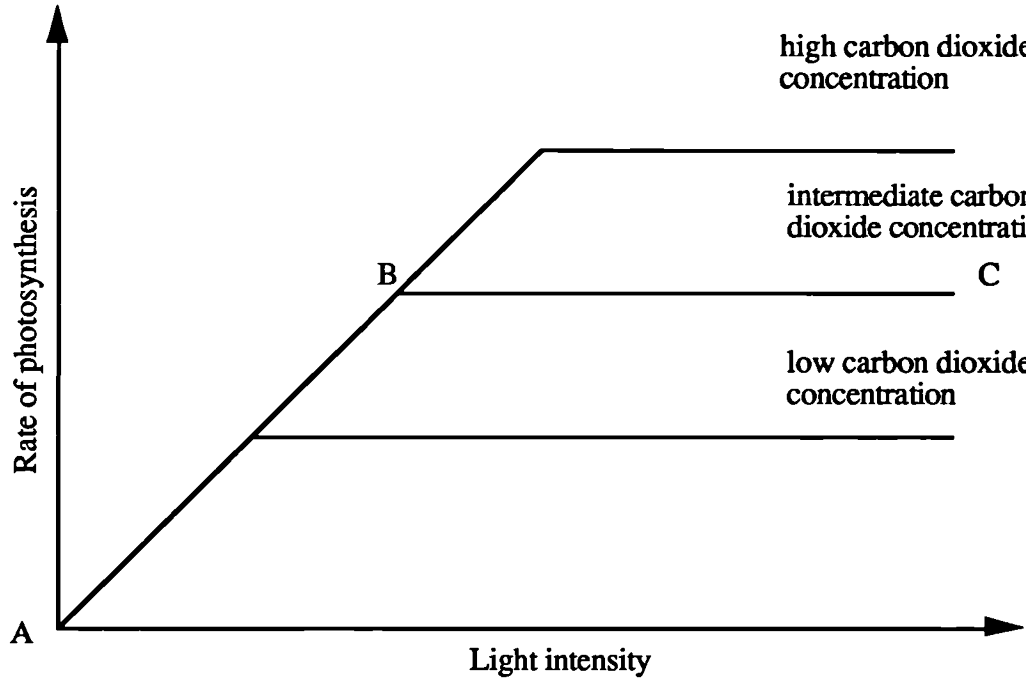


Figure 3.2 Characteristic rate of photosynthesis curves predicted by Blackman's principle of limiting factors

The difference between the form of the curves predicted by a strict application of Blackman's principle of limiting factors, and the shape of the stereotypical "Blackman" curves obtained from experimental results (and depicted in school textbooks) can be explained by an insistence that only one factor can limit the rate of photosynthesis at any one time. It is clear from the stereotypical curves shown in Figure 3.1 that at low light intensities, only light intensity is represented as a limiting factor (section AB), and that only carbon dioxide intensity is represented as a limiting factor at high light intensities (section CD). However, in the intermediate section BC, both factors are limiting the rate of photosynthesis.

An understanding of the typical curves used in textbook explanations of limiting factors is important, as the characteristics of these curves can be used to infer that photosynthesis consists of a sequence of related chemical reactions. The general shape of these curves implies that there are at least two different processes in photosynthesis as the effects of two limiting factors are evident. In fact it is possible to relate the three main processes of photosynthesis (light harvesting, energy transduction, and carbon reduction) to the effects of the principal limiting factors (Gaastra, 1969). Light harvesting and energy transduction, are affected by light intensity, but not by carbon dioxide concentration or temperature.

However, carbon reduction is strongly affected by the concentration of carbon dioxide, and only weakly by temperature. Any dependence of this process on light intensity is only indirect; light intensity may affect stomatal diffusion resistance. The biochemical processes which precede and succeed the reduction of carbon dioxide are strongly affected by temperature, but not by light or, in most cases, carbon dioxide concentration.

3.3.3 Misconceptions about limiting factors in photosynthesis

Students' conceptual difficulties in this area have been researched by Amir & Tamir (1989, 1990) Both studies involved the application of diagnostic tests designed to identify the misconceptions and alternative ideas held by 11th and 12th grade students. The results of these studies confirm the existence of commonly held misconceptions about the the concept of limiting factors.

The first study (Amir & Tamir, 1989) was based on students' interpretation of the results of an experiment in which the rate of photosynthesis of two species of clover were measured as a function of light intensity. These results are shown in tabular and graphical form in Figure 3.3.

The study consisted of two phases. In the first phase 147 students randomly chosen from 16 randomly selected schools (which participated in the 1986 biology matriculation examination in Israel) were presented the results in tabular form only and asked to provide written responses to the following questions:

A Describe the findings presented in the table.
B At a light intensity of 70 lux, is light the "limiting factor"? - give your reasons.
C Which species, A or B, will be more suitable for growing in a shaded area? - explain.

The experience gained in the first phase was used to revise these questions for use in the second phase. In the second phase 210 11th and 12th grade students were presented with the results in graphical form instead of the tabular form used in the first phase, and the wording of question B was changed to include specific references to species A and species B. In addition to the administration of the written questions to the second sample, eight students from this sample, who had demonstrated a misunderstanding of the concept of limiting factors, were interviewed to probe the origins of their misunderstanding.

Light intensity(lux)	Rate of photosynthesis*	
	Species A	Species B
10	5	5
20	9	29
30	19	46
40	28	49
50	32	54
60	42	58
70	55	60
80	72	60

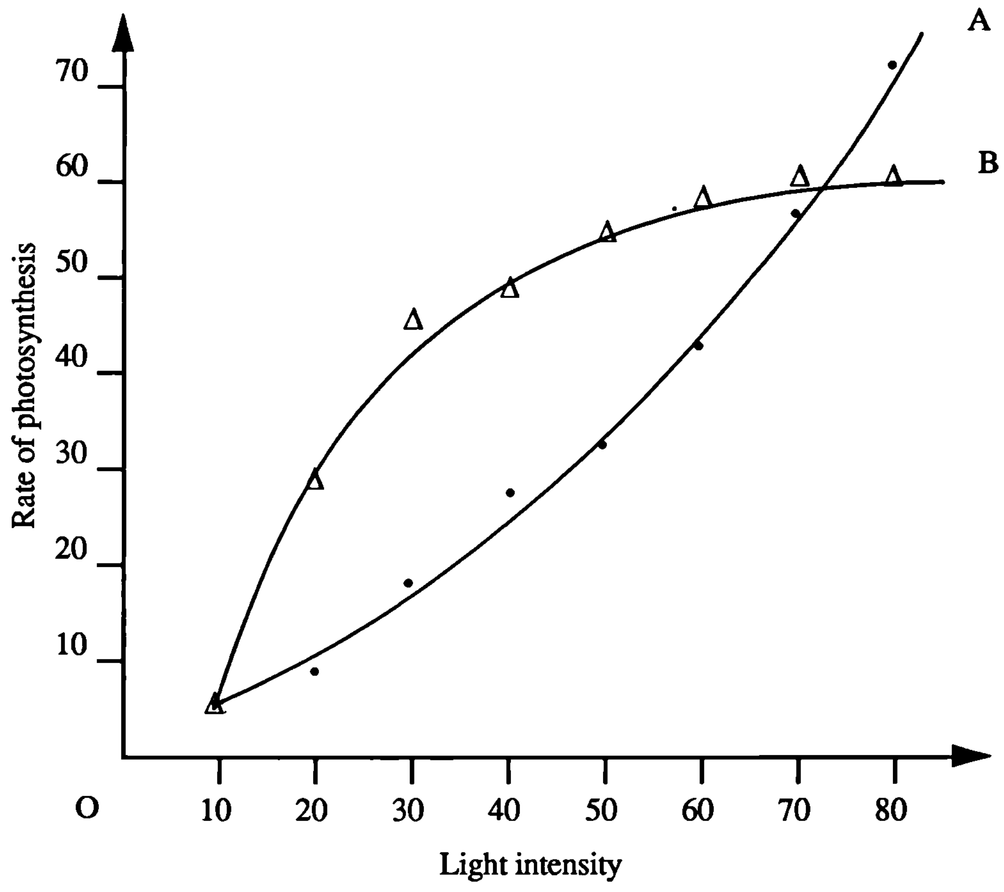


Figure 3.3: Experimental results presented to students as a basis for diagnostic tests by Amir & Tamir (1989)

Question A was intended to explore the significance of the format of data presentation. The relationships between rate of photosynthesis and such factors as light intensity and carbon dioxide concentration are usually shown in the form of a graph. In the first phase students were asked to describe this relationship with reference to tabular data, with the aim of gaining some insight into whether this form of presentation presented obstacles to understanding. Only 35% of the students successfully identified the features of the data and most students did not show that they understood the changes in the rate of photosynthesis occurring in the two species of clover. It seems reasonable to conclude that a tabular representation presents obstacles to understanding. This conclusion is supported by the findings of the second phase. Presenting the data in a graphical format appeared to help students' understanding, with more of the students giving responses which indicated an overall appreciation of the way that the rate of photosynthesis was changing with light intensity.

Question B was intended to specifically probe students' understanding of the concept of limiting factors. The responses to this question in the first phase were analysed in terms of (i) the students' identification of which species of clover is limited by a light intensity of 70 lux, and (ii) the reasons advanced for the identification of this species.

Students' identification of the species limited by a light intensity greater than 70 lux indicated the existence of widespread misunderstanding about the concept of limiting factors; only 21% of the students answered correctly that species A is limited by a light intensity of 70 lux, with 52% wrongly identifying species B, 25% providing a doubtful identification which either did not mention a species specifically or which mentioned both species, and 2% providing no identification.

The reasons given for the identification of the species limited by light intensities above 70 lux revealed the existence of a common misconception. 84% of the students gave a correct answer of one form or another; 16% answered that the rate for species B is constant between 70 lux and 80 lux, 7% answered that the rate for species A increases between 70 and 80 lux, and 61% answered with a combination of these two reasons. Given that only 21% of the students correctly identified species A, it would appear that a significant number of students are using correct reasons to justify an incorrect identification. In fact further analysis showed that 71% of the students who incorrectly identified species B used the combination of correct reasons as a justification. Amir and Tamir (1989) quote two sample answers to illustrate this point:

> *Correct answer*
> Light intensity of 70 lux is limiting the rate of photosynthesis for the species A because the species has produced more sugar when the light intensity was increased. Light intensity of 70 lux is not limiting species B because no increase in production occurred.

Incorrect answer

Light intensity of 70 lux is limiting the rate of photosynthesis for species B since it can be seen that the photosynthetic rate did not change when the light intensity was changed to 80 lux. On the other hand, for species A the rate increased when the light intensity was increased to 80 lux. This shows that for species A 70 lux was not limiting.
(Amir & Tamir, 1989, p. 131)

Students who used a correct explanation to justify an incorrect response indicate the existence of a misconception; a limiting factor operates when the rate of the process in question does not increase despite increasing the intensity of the factor.

In the second phase, the wording of question B was changed to ascertain whether distinct reference to species A and species B would help to clarify students' ideas. The revised form of question B was as follows:

B Does light of 70 lux constitute a "limiting factor" -
for species A? - Explain
for species B? - Explain

Specific mention of species A and species B appeared to help. The percentage of students giving a correct answer in the second phase (49%) was much higher than in the first phase (21%), with the percentage of students giving partially correct or doubtful answers decreasing significantly. However, the percentage of students giving a wrong answer remained about the same in both phases - 52% in the first phase and 45% in the second phase. It would appear that the rephrasing did not make any difference for students holding misconceptions about limiting factors. This view is supported by the similarity in the content of incorrect answers given in the first phase and the second phase.

The results of the second phase corroborated the findings in the first phase; about 50% of the students hold an alternative meaning for the concept of limiting factor, which Amir and Tamir (1989) express as:

> [...] a limiting factor exhibits itself in a situation where the rate of the process does not increase despite increasing the intensity (or amount) of that factor.
> (Amir & Tamir, 1989, p. 131)

Question C was intended to test students' ability to apply the concept of limiting factors to solve a simple problem. However, the responses to question C in the first phase did not give any insights into students' understanding of limiting factors. Although 95% of the students correctly identified species B as the most suitable for growing in a shaded area, most (83%) supported their answer by stating that species B had a higher photosynthetic rate at lower light intensities. Only 3% of the students used the concept of limiting factors to support their answer. The results for Question C in the second phase were almost identical.

During the interview sessions in the second phase the eight selected students were shown the graph of the experimental results for species A and B, and asked the following questions: (i) What is a limiting factor? (ii) When can light be a "limiting factor"? and (iii) Why do you think your answer to question B was wrong? The answers to the first of these questions demonstrated partial understanding with the frequent use of analogies. Whilst the analogies were not in themselves incorrect, the answers to the second question revealed that the students held the misconception previously revealed by an analysis of the written answers. In addition, a different misunderstanding was shown by some interviewees; changes in the rate of photosynthesis were ignored and the concept of limiting factor was viewed in terms of the light intensity, with some interviewees stating that as the light intensity is increasing it cannot be a limiting factor. The answers to the third question provided some insights into the sources of students' misunderstanding, and Amir and Tamir (1989) list illustrative examples of statements from interviewees such as "I thought light was not limiting because the graph continues to rise", and "Limit means "stop", that's why I gave a wrong answer". The results of the interview sessions confirm the findings from the written responses, and Amir and Tamir (1989) conclude:

> A 'limiting factor' is grasped by the students as something which when abundant prevents an increase in the rate of the process. The levelling off of the graph, the decrease in rate (at high light intensities) is wrongly interpreted to mean that a limiting factor is affecting the process. (Amir & Tamir, 1989, p. 133)

The second study (Amir & Tamir, 1990) involved the use of two diagnostic items with a sample of 285 11th and 12th grade students. The first diagnostic item consisted of a multiple choice question which required comprehension and analysis. This item showed a graph of the results of an experiment in which the rate of photosynthesis was measured at different light intensities, as shown in Figure 3.4. The students were asked to choose one of the options in the following question and to give a brief justification of their choice:

By examining the rate of photosynthesis (shown in the graph) it may be inferred that at light intensities higher than 3000 foot-candles:

a photosynthesis has ceased

b the rate continues to rise with the increase in light intensity

c light has become the limiting factor

d another factor acts as the limiting factor

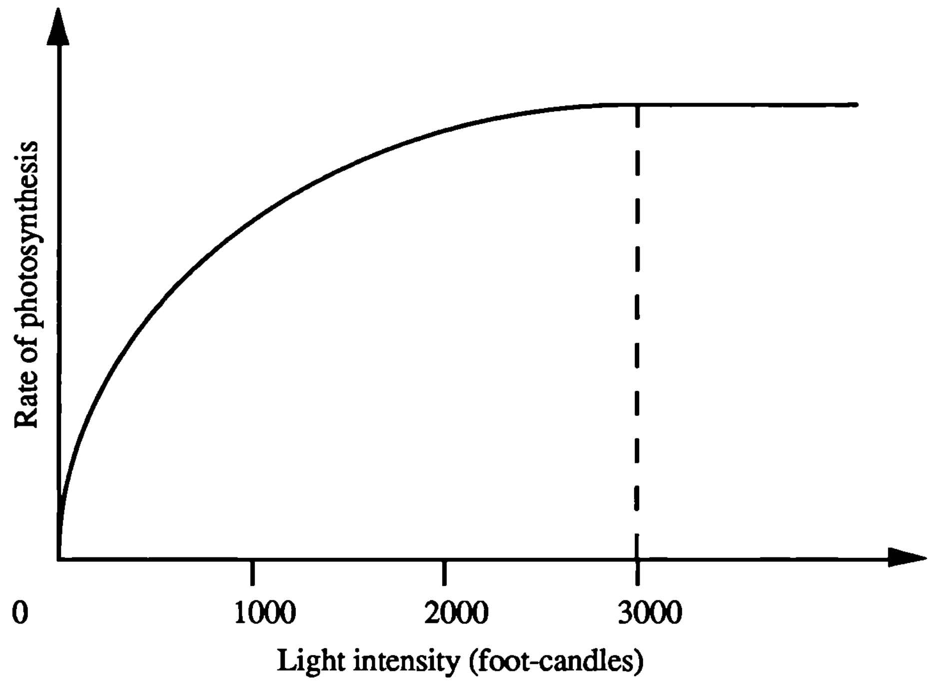


Figure 3.4: Graph used in the multiple choice question featured in the study by Amir & Tamir (1990)

The answer to this question appeared to indicate that most students understood the concept of limiting factors. 81% of the students gave the correct response of "d". 17% gave "c" as their response, indicating that 98% of the sample realised that the question concerned the concept of limiting factor. However, as Amir & Tamir point out, this is not surprising as most textbooks introduce the concept of limiting factors with reference to graphs very similar to that used in the multiple choice question. The 17% who chose the distractor "c" showed clear evidence of holding the same misconception identified by Amir & Tamir (1989); namely, that a limiting factor operates when the rate of the process in question does not increase despite increasing the intensity of the factor.

An analysis of the justifications provided by students revealed a less impressive level of understanding. Justifications were judged to be correct if they included one or both of the following: (i) which factor might be the limiting factor, and (ii) adding to (or increasing) the limiting factor will bring about an increase in the rate. On this basis only 55% of the students provided a correct answer, with 27% providing a partial answer based on the observation that the graph levels off at high light intensities, and 18% giving an incorrect answer that light is limiting at high light intensities.

The second item was the revised form of question B used in the second phase of the study conducted by Amir & Tamir (1989). The results obtained from an analysis of the responses to this item were in broad agreement with the results of the use of this item in the first study (Amir & Tamir 1989). In this second study 42% of the students correctly identified species A as being limited by a light intensity of 70 lux, compared to 49% in the first study; and 56% incorrectly identified species B in the second study, as compared to 45% in the first study. Amir & Tamir conclude from the results of their second study:

> [...] although many students can identify a correct statement about a limiting factor, about half of them do not fully understand the concept. Many students believe that a limiting factor is inhibiting or slowing down the process when its amount or intensity is high rather than at a minimum. (Amir & Tamir, 1990, p.9)

The results of the two studies by Amir & Tamir provide evidence of students' misunderstanding of the concept of limiting factors, with a clear indication of a common misconception that a limiting factor limits the rate of photosynthesis when the saturation rate for a factor has been reached, rather than when an increase in the factor results in an increase in the rate. The results of the interviews conducted in the second phase of the first study also indicate the existence of another misconception; namely, that a factor cannot be limiting if it is clear (in this case from observation of a graph) that the value of the factor is increasing. Students' responses to the data presentation format diagnostic item administered in both phases of the first study indicate that the form of data presentation is important; greater understanding was shown when data were presented in a graphical form, as opposed to a tabular form.

3.4 Summary

The literature shows that students have significant conceptual problems in understanding photosynthesis, including misconceptions and alternative conceptions associated with the biochemical processes of photosynthesis. One aspect of these biochemical processes, the interaction between the principal environmental factors that affect the rate of photosynthesis, provides an appropriate task domain for the study of the educational design of *Bioview*. The three interacting variables of light intensity, carbon dioxide concentration, and temperature, which are typically considered to be the principal limiting factors, can be represented by the datacube metaphor, as in the PSYNTH datacube provided with the *Bioview* package (see Section 4.1). The interactions between the factors can be explored by manipulating the datacube and inspecting the resulting graphical representations.

The topic of limiting factors is particularly suited to research into the educational benefits of using *Bioview*. There is clear evidence that students have difficulties in understanding this concept; a fact which is emphasised by Wandersee (1988):

> In general, students' responses [...] gave little evidence of a sound understanding of the law formulated by Liebig, Mitscherlich, and Blackman. Perhaps science educators should consider the implications of such a finding in teaching the concept of photosynthesis. (Wandersee, 1988, p. 164)

As Wandersee points out there is a need to consider the implications of these difficulties. Educational software which has been designed to take account of (i) research into students' understanding and (ii) experience of teaching this concept, may be worth considering in this context. Specifically the use of *Bioview* may help in both of these respects. The findings of Amir and Tamir (1989) indicate that the concept of limiting factors is best understood through the use of graphs, which is a prominent feature of design of *Bioview*. Most textbooks provide explanations of photosynthesis which use two dimensional graphs. The three dimensional representation of the datacube and the animation of conventional two-variable graphs with respect to a third interacting variable described in Chapter 4 may afford an effective three dimensional representation.

Chapter 4

Bioview: a direct manipulation software tool

Bioview is a direct manipulation educational software package designed to allow the exploration of the relationships between three interacting variables. The program runs in the Microsoft *Windows* operating system. Direct manipulation is based on the use of a "datacube"; a three dimensional pictorial representation of the values associated with the three interacting variables. It is possible to use different datacubes, which have either been created by the user or supplied with the package. This research features the use of the PSYNTH datacube; a datacube supplied with the package which represents the rate of photosynthesis for three principal factors which interact to limit this rate. In this chapter the direct manipulation features of *Bioview* and relevant aspects of *Windows* are described. A full description of the package is given in Appendix 1.

4.1 *Bioview* as a direct manipulation tool

The three principal interacting factors represented in the PSYNTH datacube are the atmospheric temperature, the carbon dioxide concentration, and the light intensity. It is straightforward to envisage how the relationship between any two of these factors could be represented by a datasheet in which the rate of photosynthesis values corresponding to specific values of one variable are shown as rows and values corresponding to the other variable are shown as columns. For example, a datasheet for values of the rate of photosynthesis for light intensity and carbon dioxide is shown in Figure 4.1. In this datasheet, rate values for a specific value of carbon dioxide (C-value) are shown as a row, and rate values for a specific value of the light intensity (L-value) are shown as a column. These C- and L-values correspond to a fixed value for the temperature (T-value). The intersection between a row and a column corresponds to the value for a specific C-value and a specific L-value.

The datasheet representation suggests possibilities for the direct manipulation of the values associated with the two variables. By "moving" the position of the row on the datasheet the C-value will be changed. Similarly, by moving the position of the column the L-value will be changed. In a computer-based datasheet it would be possible to provide an operation which could be executed to move a row (m_row operation) and and a operation which could be executed to move a column (m_col operation).

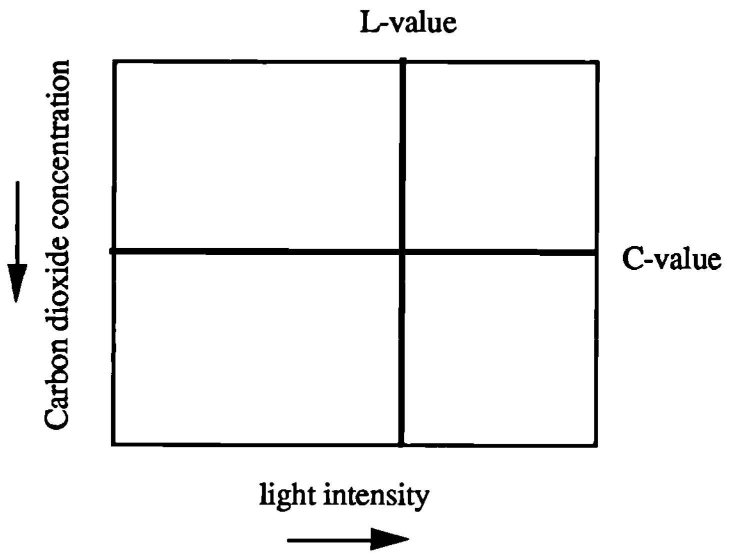


Figure 4.1: Datasheet for light intensity and carbon dioxide levels.

If the temperature is changed, another datasheet corresponding to a new T-value will be produced. If the temperature is changed for a second time, a third datasheet will be produced. In fact it is possible to imagine a series of datasheets, with each sheet corresponding to a fixed T-value. This series of sheets could be represented as shown in Figure 4.2.

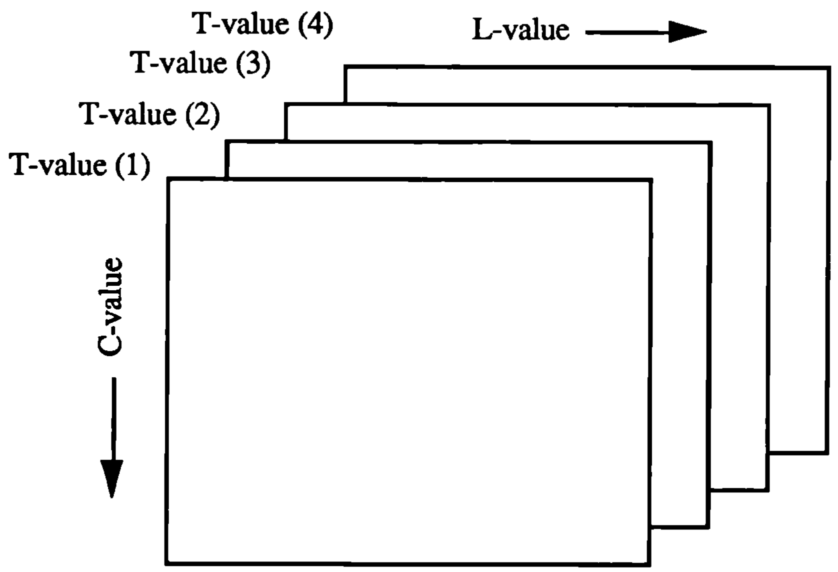


Figure 4.2: A series of datasheets corresponding to different T-values.

The series of sheets shown in Figure 4.2 can be thought of as a "datacube" as shown in Figure 4.3. In this representation temperature is now a third variable and selecting a T-value results in the selection of a datasheet representing values for the rate of photosynthesis corresponding to various C- and L-values at this T-value.

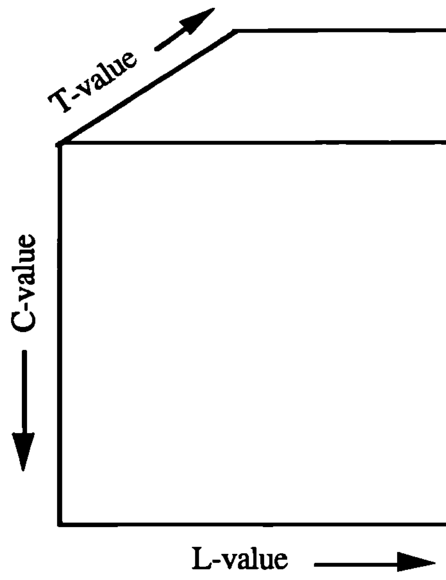


Figure 4.3: The PSYNTH datacube for factors affecting the rate of photosynthesis

The datacube can also be thought of as a series of L-sheets and C-sheets as shown in Figure 4.4 and Figure 4.5.

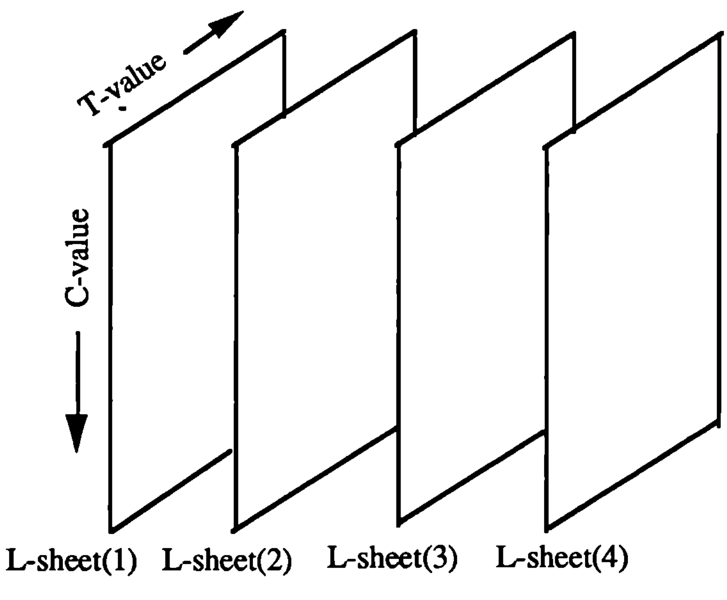


Figure 4.4: The PSYNTH datacube viewed as a series of L--sheets

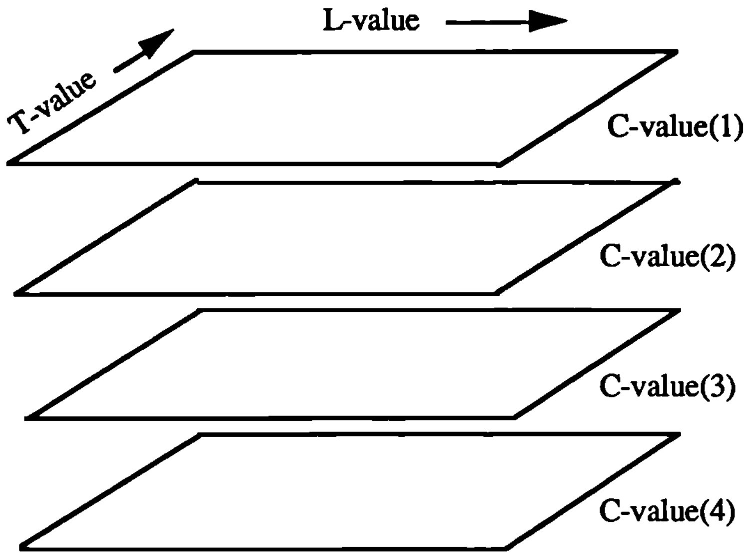


Figure 4.5: The PSYNTH datacube viewed as a series of C-sheets

Thus the datacube consists of three sets of commonly orthogonal datasheets, with each set corresponding to a fixed sheet value for one of the three interacting variables.

In a datacube, the value of the rate of photosynthesis corresponds to the value of three variables; not two other variables as in a datasheet. This is illustrated in Figure 4.6.

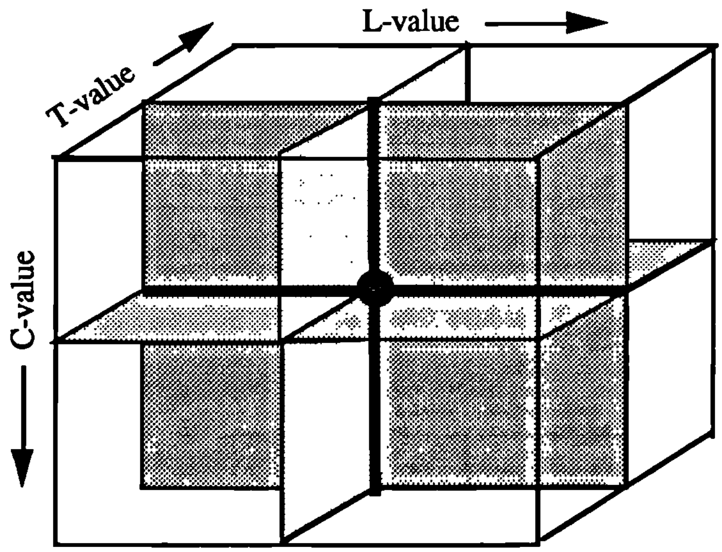


Figure 4.6: How the PSYNTH datacube defines variable values

The variable values associated with the rate corresponding to the marked position can be defined in terms of the variable value associated with one of the three orthogonal datasheets and the row and column values associated with the row and column for this sheet that pass through the marked position. For the PSYNTH datacube the variable values for a rate value are defined in terms of the values associated with (i) a L-sheet, C-row, and T-column, (ii) a C-sheet, T-row, and L-column, or (iii) a T-sheet, C-row, and a L-column.

The value of the sheet variable can be changed by executing an m_sheet operation, and the values of the variables on a selected sheet can be changed by executing m_row or m_col operations. Hence the direct manipulation operators which can be applied to the datacube are the three m_sheet operators. For the PSYNTH datacube these operators will be:

- m_sheet(L) which can be executed to change the position of the L-sheet, resulting in a change in the L-value associated with the L-sheet

- m_sheet(C) which can be executed to change the position of the C-sheet, resulting in a change in the C-value associated with the C-sheet

- m_sheet(T) which can be executed to change the position of the T-sheet, resulting in a change in the T-value associated with the T-sheet

The direct manipulation operators which can be applied to a datasheet are an m_row operation and an m_col operation. These operators for each of the datasheets in the PSYNTH datacube are:

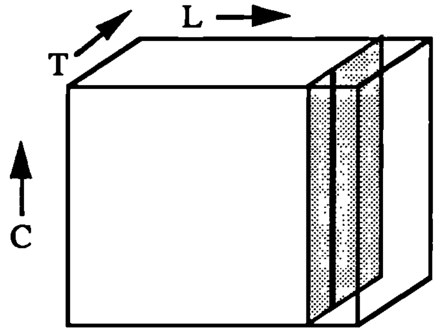
- m_row(C) and m_col(T) for the L-sheet, which can be executed respectively to change the C-value and the T-value for the L-sheet

- m_row(T) and m_col(L) for the C-sheet, which can be executed respectively to change the T-value and the L-value for the C-sheet

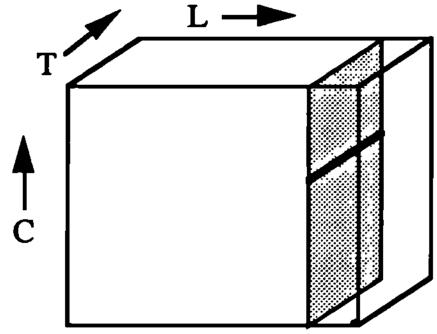
- m_row(C) and m_col(L) for the T-sheet, which can be executed respectively to change the C-value and the L-value for the T-sheet

In *Bioview* the values for each datasheet are lexically scoped, that is, a change in the values associated with a given sheet does not result in a change in the values associated with the other two sheets. For example, an m_sheet(C) operation will only change the C-value associated with the C-sheet, leaving the C-values for the T- and L-sheet unchanged.

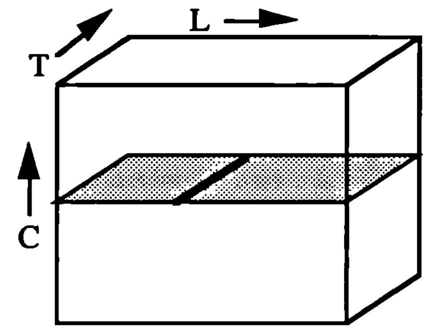
Any two of the three "move" operators associated with a datasheet can be executed in sequence to change the value of two variables while the value of the third variable remains fixed, giving nine pairs of operators that can be applied to change the values of two of the variables. For the PSYNTH datacube the six pairs of m_sheet and m_row/col operators are (i) an m_sheet(L) with a m-col((T) operator, (ii) an m_sheet(L) operator with m_row(C) operator, (iii) an m_sheet(C) operator with m-col((L) (iv) an m_sheet(C) operator with an m_row(T) operator, (v) an m_sheet(T) operator with an m_col(L) operator, and (vi) an m_sheet(T) operator with m-row(C) operator. These pairs are shown in Figure 4.7. In addition there are three pairs of m_row(C/T) and m_col(L/T) operators; one pair for each datasheet.



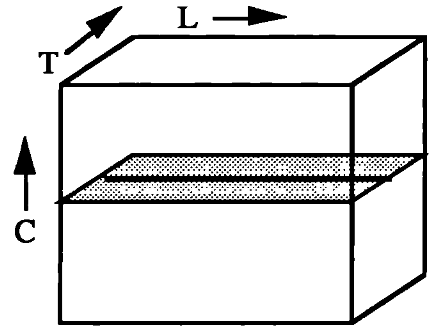
(i) L-sheet and T-column selected



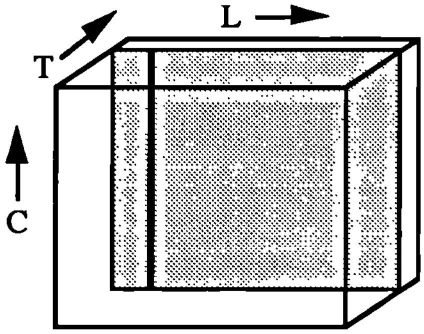
(ii) L-sheet and C-row selected



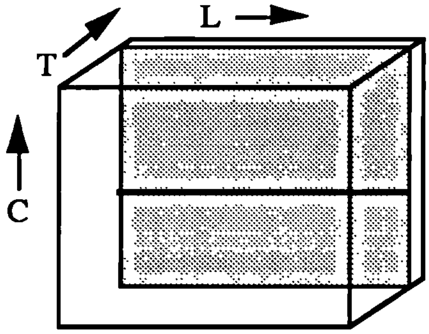
(iii) C-sheet and L-column selected



(iv) C-sheet and T-row selected



(v) T-sheet and L-column selected



(vi) T-sheet and C-row selected

Figure 4.7: Sets of direct manipulation operators for the PSYNTH datacube.

In *Bioview* the datacube and datasheet are provided as *Windows* direct manipulation widgets, with the datacube shown in a window and the currently selected datasheet shown in another window. The value of the variable associated with a chosen datasheet can be changed by directly manipulating the datacube widget. It is possible to select any of the three datasheets by clicking on the appropriate datacube face or value scroll-box, and to change the position of this sheet by clicking or dragging the slider associated with the selected sheet. The variable associated with a row or column in the selected datasheet can be selected by clicking on the appropriate row or column displayed in the datasheet widget. The selected row or column becomes "highlighted", that is displayed in reverse video, and the variable associated with the selected row or column can be changed by clicking on a new row or column or dragging the position of the row or column across the datasheet.

From the discussion in this chapter it is clear that the design rationale of *Bioview* is based on the direct manipulation paradigm. However, the design has made allowance for the users who do not wish to use the datacube representation to select a datasheet. The "View" pull-down menu item in the datasheet window allows users to select a datasheet by selecting from a menu which lists each of the datasheets. However, once a datasheet has been selected in this way the sheet, row and column variable values can only by changed by executing the direct manipulation m_sheet, m_row and m_col operations.

The novel form of exploration that direct manipulation of the datacube affords can be illustrated by directly manipulating the PSYNTH datacube. This datacube contains results which produce Blackman type curves for the rate of photosynthesis. When light levels are high a low level of carbon dioxide limits the rate of photosynthesis until a 0.5% of carbon dioxide concentration is reached. The rate of photosynthesis increases with temperature to 30 ^{o}C and then decreases towards 40 ^{o}C. When light levels are high and the carbon dioxide concentration is normal, the rate is only slightly affected by temperatures between 15 ^{o}C and 30 ^{o}C.

The opportunity to manipulate the PSYNTH datacube with *Bioview* enables stereotypical Blackman type curves to be studied in a far more flexible way than is possible with conventional graphical representations. The variations of rate of photosynthesis with principal limiting factors can be represented by various types of rate/variable graphs (see Section 3.3.2); an approach which is supported by the findings of Amir and Tamir (1989). However, these graphs can also be linked in animated sequences by manipulating the datacube, offering novel ways of exploring the effects of limiting factors on photosynthesis.

A specific example of a novel use of *Bioview* in understanding the effects of limiting factors is given by considering the suggestion by Heath (1969) that a full understanding of the interactions between the principal limiting factors in photosynthesis is assisted by close inspection and study of a three dimensional isometric representation of the simultaneous effects of changing light intensity and carbon dioxide concentration on the rate of photosynthesis, as shown in Figure 4.8.

Bioview offers an alternative approach to exploring the interaction between these factors. For example, moving a selected row and a selected column through a datasheet corresponding to a fixed temperature, and viewing the resulting animated sequence of graphs of rate against light intensity and carbon dioxide concentration provides an alternative dynamic representation to the static representation suggested by Heath.

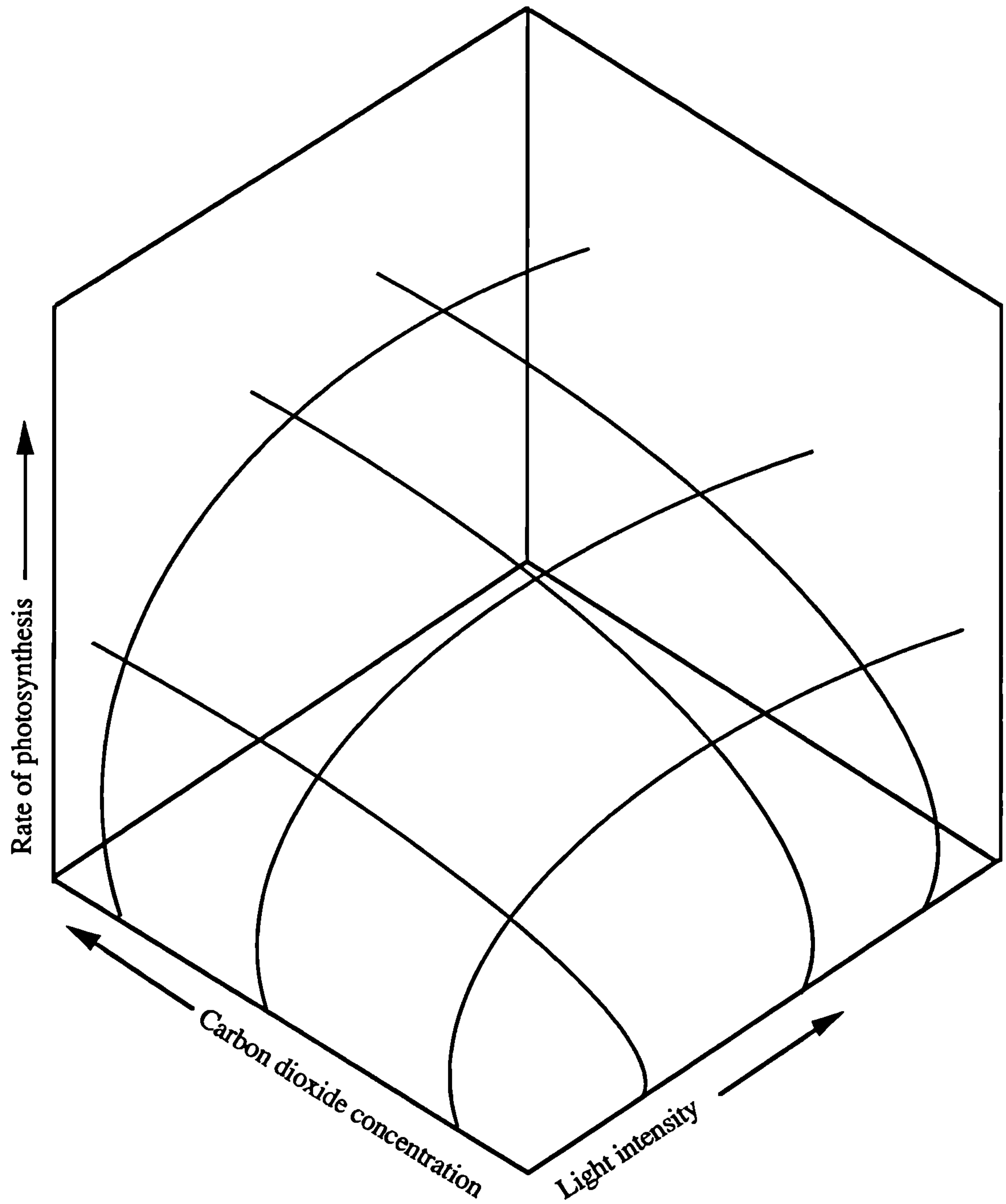


Figure 4.8: Three dimensional isometric representation of the simultaneous effects of light intensity and carbon dioxide concentration on the rate of photosynthesis

4.2 *Bioview* as a *Windows* application

The implementation of *Bioview* in Microsoft *Windows* has a number of consequences for the user. Firstly, in line with the *Windows* rationale, the interface for *Bioview* consists of multiple interacting windows which can be positioned, sized, minimised, and maximised at the discretion of the user. Secondly, the multi-tasking nature of *Windows* has been utilised to link the windows in such a way that actions executed in one window automatically affect the other windows as well as the action target window. Thirdly, it is possible to have other applications in addition to *Bioview* running in windows shown on the screen. These *Windows* related consequences will be considered in this section.

4.2.1 Multiple windows

A *Bioview* user can display the following types of window: a datacube window that provides a pictorial display of the datacube; a datasheet window that displays all or part of the datasheet, depending on the relative size of the datasheet; an analysis window that provides a simple statistical analysis of the data currently highlighted in a row or column; and multiple graph_windows that can be used to display graphs of the rate of photosynthesis plotted against the variable corresponding to the currently highlighted row or column. Only one graph_window at a time can be "connected" to the datacube and datasheet windows. There is clearly scope for the user to produce potentially confusing screen displays. Parts of a window which shows vital information or which is required to directly execute an operation could inadvertently be obscured by overlapping windows. Also some windows may be minimised at inappropriate times, making some direct manipulation operations impossible or leading to the non-display of important information. The possibility of developing confusing screen displays is recognised through the provision of a Tidy option which returns the position of the datacube, datasheet, analysis, and first graph window to their initial default positions.

The datacube window is central to the design of *Bioview*. The display of the datacube in this window provides a navigational framework for using the package to explore the relationships between interacting variables. In addition, it is through the direct manipulation of the datacube that datasheets are selected and the variable value associated with a datasheet is changed. If this window is obscured by other windows, or minimised to an icon, there are obvious implications for the effective use of the program.

Constraints imposed by the size of the screen display mean that in most cases only a part of the datasheet will be displayed in the datasheet window. Maximising the datasheet window will result in the display of the whole sheet, unless the datasheet is exceptionally large. However, a maximised sheet will take up the whole screen, making it impossible to see the effect on the graph in the connected graph_window of directly manipulating the sheet through the execution of m_row and m_col operations. When only a part of the datasheet is displayed it will be necessary to change the part displayed to explore the full range of the effects of m_row and m_col operations. This can be done by executing *Windows* scope_win operations by clicking on the appropriate window scroll box located in one of the four corners of the datasheet window. For large datasheets this can adversely affect the smooth direct manipulation of the datasheet.

The analysis window is probably the least important window in the effective use of *Bioview*. Whilst the information displayed in the window may be useful, it is probable that the effect of directly manipulating either the datacube or the datasheet will be explored by observing the changes in a connected graph window. The statistical analysis will typically be used to confirm the results of conclusions reached by inspecting the changes in the rate/variable graphs. In addition, it is not possible to execute any direct manipulation operation in this window. If this window is obscured or minimised, the consequences are not likely to be serious.

When *Bioview* is loaded one graph-window is displayed on the screen. However, consistent with memory constraints, it is possible to have multiple graph_windows displayed. A new graph is started by executing a start_graph operation. By executing a con_graph operation one of the displayed graph windows will be connected to the datasheet and datacube windows and become sensitive to changes in the state of the datacube. The other graph windows displayed on the screen will be "frozen", and will not be affected by the execution of operations in the datacube and datasheet windows until a con_graph operation is executed to connect one of them. If this is done, the previously connected graph_window will be frozen.

4.2.2 Multi-tasking

The *Windows* clipboard can be used to copy output in a *Bioview* window to a window on the screen that is supporting the use of another application. For example, instances of graphs displayed in a graph_window can be copied into a word-processor However, *Windows* also allows dynamic data exchange (DDE) between the applications running in the various windows displayed on the screen. This means that the data generated by user actions targeted on one of the displayed windows can be automatically passed to applications running in other windows as the actions are completed by the user. After data has been passed to other windows they can be used by the applications in the other windows. The delay between receipt of data and its use will be imperceptible and the system will appear to be multi-tasking, that is it will seem that the applications are running at the same time with the same inputs.

The multi-tasking nature of *Windows* is utilised in the design of *Bioview*. The dynamic data exchange between *Bioview* windows is represented in Figure 4.9.

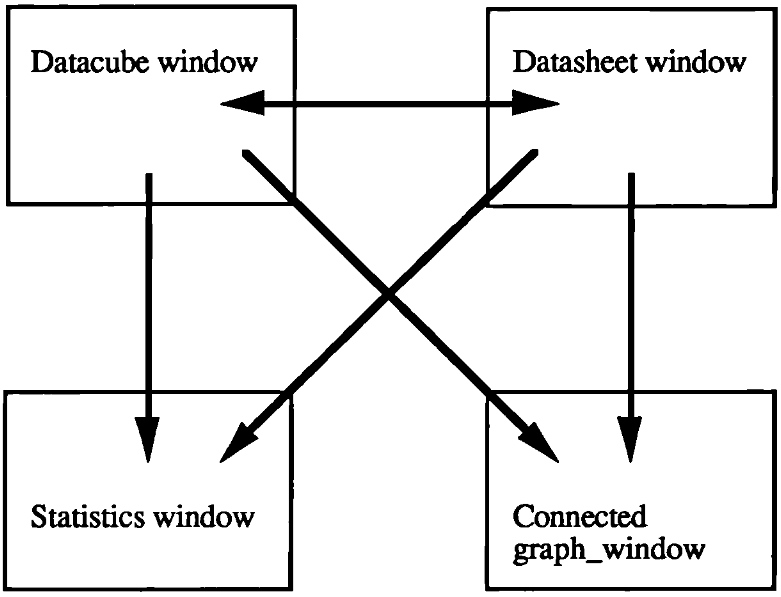


Figure 4.9: Multi-tasking features of Bioview

Data pass to and from the datacube and datasheet windows. Thus direct manipulation of one of these windows results in a corresponding effect in the other window. For example, if an m_sheet(T) operation is executed to move the position of the datasheet shown in the datacube window the datasheet shown in the datasheet window will change to the sheet corresponding to the new T-value position shown in the datacube window. Similarly if an s_sheet(C) operation is executed the datasheet shown in the datasheet window will change to a C-sheet. Changes in the position of a row or column effected by executing m_row or m_col operations will be shown in the datacube window by changes in the position of the line on the datacube which represents the current row or column position.

Data only pass from the datacube or datasheet window to the analysis window and the connected graph window. Operations executed in the analysis window or the connected graph window have no effect on the state of the datacube or datasheet window. Direct manipulation of the datacube or datasheet will result in the display of a different set of statistics in the analysis window and a change in the instance of the graph shown in the connected graph window. For example if a C-row on a T-sheet is currently selected and an s_col(L) operation is executed the displayed instance of the graph of rate of photosynthesis plotted against light intensity (rate/L graph) will be changed to a rate/C graph.

The link between the connected graph window and either the datacube window or the datasheet window can be used to generate sequences of instances of graphs. For example, the execution of a series of m_sheet(T) operations with a C-row selected will result in the display of a sequence of rate/L graphs, with each graph corresponding to a different T-value. Executing the m_sheet operations by successively clicking on the scroll box for the sheet variable value will result in a series of "snapshots" of instances of the graph. Dragging the sheet value slider will result in a rapid succession of snapshots creating an animated display of changes in the form of the graph. Similar effects can be achieved by clicking in succession on row or column positions or dragging the position of a row or column.

4.3 Representing direct manipulation in *Bioview*

As discussed in Chapter 2 an enhanced version of the GOMS model will be used to represent the human-computer interaction when *Bioview* is being used. Given this decision it should be possible to (i) identify a set of operators which can be used to manipulate each of the *Bioview* windows, (ii) describe the interaction in terms of methods designed to achieve given tasks, and (iii) identify rules which can be applied to select an appropriate method.

4.3.1 *Bioview* operators

As the datacube and datasheet widgets can be directly manipulated they have direct manipulation operators associated with them. The graph widget cannot be directly manipulated but the graph_windows have a number of configuration operators associated with them. The analysis window does not have any direct manipulation or configuration operators associated with it. The operators associated with each widget are shown in Table 4.1. The sheet that an operator is applied to is indicated in brackets as as a L, C, or T. Graph types are indicated as l (line graph), b (bar graph) or p (pie chart). Some of these operators have not been described in this chapter. Descriptions of these operators are given in Appendix 1.

The "desk-top" can also be directly manipulated in the usual way by using *Windows* operators. A window can be iconised by executing a minimise operation or expanded to fill the whole screen by executing a maximise operation. The position of a window can be changed by dragging it to a new position by executing a move_window operation. A window's size can be altered by dragging the edge or corner of a window to execute a size_window operation. The scope of a window can be changed by executing a scope_win operation. This can be done by clicking on one of the four window scroll boxes located at each of the four corners of the window or by dragging the horizontal or vertical sliders. The use of scope_win operators can be critical to the effective direct manipulation of the datasheet as explained in Section 4.2.1.

Table 4.1: Bioview operators

	Widget		
Function	Datacube	Datasheet	Graph_window(x)
select a L-sheet	s_sheet(L)		
select a C-sheet	s_sheet(C)		
select a T-sheet	s_sheet(T)		
move a L-sheet	m_sheet(L)		
move a C-sheet	m_sheet(C)		
move a T-sheet	m_sheet(T)		
select a L-row		s_row(L)	
select a C-row		s_row(C)	
select a T-row		s_row(T)	
move a L-row		m_row(L)	
move a C-row		m_row(C)	
move a T-row		m_row(T)	
select a L-column		s_col(L)	
select a C-column		s_col(C)	
select a T-column		s_col(T)	
move a L-column		m_col(L)	
move a C-column		m_col(C)	
move a T-column		m_col(T)	
connect a graph_window			con_graph(x)
launch a graph_window			start_graph(x)
select a line graph			s_graph(l)
select a bar graph			s_graph(b)
select a pie chart			s_graph(p)
scale a graph			scale_graph
inspect a rate/L graph			inspect_graph(r/L)
inspect a rate/C graph			inspect_graph(r/C)
inspect a rate/T graph			inspect_graph(r/T)

4.3.2 *Bioview* methods

It is possible to identify a set of successful methods that can be applied to explore the relationships between the three variables represented in the datacube. Three types of basic methods have been identified. In Method 1 the value of one variable is set to a required value by executing an m_sheet operation, the value of a second variable is set to a required value by executing an m_row or m_col operation, and the variation in the third variable observed by the inspection of a connected graph. Hence in this method direct manipulation is used to fix the values of two variables and the variation of the third variable is observed by the "indirect" conventional inspection of a graph. In an application of Method 2 the value of one variable is fixed by executing an m_row or m_col operation, the value of the second variable is fixed by focusing on a specific value shown on the horizontal axis of the graph shown in a connected graph_window, and the third variable is varied by executing m_sheet operations. The effect of varying the third variable is observed by inspecting the changes in the graph displayed in the connected graph_window. If the full range of the horizontal axis of the displayed graph is observed, rather than focusing on a specific value on the axis, the user is observing the changes in two variables at the same time; one by observing the variation in the displayed graphs as the sheet is moved, and one by inspecting each instance of the graph in a indirect fashion. Method 2 uses direct manipulation to fix the value of one variable and vary the value of another variable. In Method 3 the value of one of the variables is again fixed by focussing on a value shown on the horizontal axis of a connected graph. However, this time the value of one of the other variables is fixed by the sheet position, and the value of the remaining variable is varied by executing m_row/col operations. Of course, as in Method 2, the user has the option to consider the full range of values shown on the horizontal axis. Direct manipulation is used to fix the value of one variable and vary the value of another variable, as was the case in Method 2. Each of these methods can be applied to each of the three datasheets. The applicable data sheet is indicated in brackets after the method number, for example Method 3(T) represents Method 3 applied to the T-sheet.

The form of a basic method can be illustrated by considering Method 3(T). It is possible to vary either the C-value by executing m_row(C) operations or the L-value by executing m_col(L) operations. It is assumed that the user wishes to vary the L-value. The user applying this method will select a T-sheet and move the sheet to the position corresponding to the required temperature, select a L-column, focus on a specific C-value on the horizontal axis of the rate/C graph displayed in the connected window, and move the L-column and observe the changes in the chosen C-value as the rate/C graph changes. This method can be represented as the following "unit-tasks":

Unit-task 1: Decide on a variable (temperature) to fix at a given value and set the variable to the required value. (|| s_sheet (T) | m_sheet(T))

Unit-task 2: Decide on a variable whose value will be varied (light intensity) and select a column corresponding to this variable. (|| s_col(L))

Unit-task 3: Decide on a variable (carbon dioxide level) to fix at a given value and locate this value. (|| inspect_graph(r/C))

Unit-task 4: Vary the value of the light intensity and observe the effect on the rate of photosynthesis. (|| repeat [m_col(L) | inspect_graph(r/C)])

The method can be represented by the following action string:

|| s_sheet(T) | m_sheet(T) || s_col(L) || inspect_graph(r/C) || repeat [m_col(L) | inspect_graph(r/C)]

The boundaries between unit tasks, for example between selecting the T-sheet and locating it at a position corresponding to the required temperature, are marked by "||" markers, and the boundaries between individual operators are separated by "|" markers.

If it is intended to compare the effects of varying two variables, basic methods can be combined in sequence to form "double" methods. For example, an application of Method 2(T) with a C-row selected followed by an application of Method 3(T) with a L-column selected could be used to compare the effects of changing the temperature and changing the light intensity. Action strings for the 18 possible successful single and 153 successful double methods are given in Appendix 2.

Successful methods can be truncated due to task requirements or the state of the system when the application of the method is commenced. Both forms of truncation can be illustrated by considering any of the successful methods identified in Appendix 2, such as Method 3(T).

Task related truncation arises from the decision to fix the value of one, rather than two, of the interacting variables. In its full form this method is designed to explore the the effect of changing the value of one variable while the other two variables are fixed. One variable is fixed by the sheet position and the other variable is fixed by the the selection of a specific value on the horizontal axis of the rate/variable graph which is currently connected. If only one variable is to be considered as fixed, a specific value on the horizontal axis of the connected graph does not need to be selected. This would make the inspect_graph(r/L) operation in Unit-task 3 redundant and a task truncated version of Method 3(T) would have the following task-truncated action string:

|| s_sheet(T) | m_sheet(T) || s_col(L) || repeat [m_col(L) | inspect_graph(r/C)]

Display related truncation can arise in a number of ways. If columns are highlighted when the T-sheet is selected the action string for the task-truncated method will become:

|| s_sheet(T) | m_sheet(T) || repeat [m_col(L) | inspect_graph (r/C)]

Further partial truncation would result from a T-sheet already being selected at the start of the implementation of the method, resulting in the following action string:

|| m_sheet(T) || repeat [m_col(L) | inspect_graph (r/C)]

If the T-sheet was already located at the desired temperature position Unit-task 1 would become unnecessary and the action string would be fully truncated:

|| repeat [m_col(L) | inspect_graph (r/C)]

The action string length for a truncated method is simply determined by counting the number of operators in the truncated string.

4.3.3 Selecting *Bioview* methods

The GOMS methodology is based on the notion of users selecting what they feel is the most appropriate method in a given situation, and that an individual user's selections can be expressed in terms of selection rules. It is probable that selections are influenced by what methods are considered to match most effectively the nature of the task in hand, the ease with which a method can be applied, or both. Task related selection could relate to the type of graphical representation implicit in a method. For example, if it is desired to determine the specific value of a variable, it may be that a user would prefer to use a method that featured the inspection of a static rate/variable graph rather than an animated sequence of rate/variable graphs. The ease with which a method can be applied may well depend on the current state of the system. If a T-sheet is currently displayed the application of Method 3(T) may well be preferred to the application of Method 2(L) to explore the effect of changing the light intensity. Method selection is considered in detail in Section 7.3.3.

4.4 Summary

Direct manipulation in *Bioview* is based on the notion of a datacube; a three dimensional representation of three interacting variables. The datacube, and associated datasheets, can be directly manipulated to change the values of the the three interacting variables. The effects of changing the values of variables can be observed by inspecting the changes in rate/variable graphs corresponding to either the rows or columns of a datasheet.

It is possible to identify a limited number of operators which can be executed in order to effect direct manipulation. Sequences of operators can be described as methods which can be applied to specific tasks. Typically more than one method is applicable to a given task. Method selection is probably based on both task and display considerations, leading to the possibility of task and display related selection rules.

Chapter 5

Research Methodology

Nielsen & Molich (1990) maintain that there are four basic ways to evaluate a user interface: (i) an analysis technique based on the application of a formal model of human-computer interaction, (ii) a computerised procedure to automatically record user actions and system responses, (iii) an empirical study based on experimental observations, and (iv) an heuristic evaluation by simply using an interface and passing an intuitive judgement on its design. There are problems in applying each of these techniques to the evaluation of direct manipulation software tools. As discussed in Chapter 2, established formal models of human computer interaction, for example GOMS, tend to ignore system output. Without modification this makes them inappropriate as representations of human-computer interaction in direct manipulation environments. An automatic record of human-computer interaction will typically be at the keystroke or mouse-click level, providing, as Dix et al. (1993) describe it, a "semantic free" data collection technique. Given the context rich environments of most direct manipulation interfaces, this approach has obvious limitations. Springett & Grant (1993) claim that while empirical evaluation is a very good technique for identifying design problems, it is not very good at identifying solutions to these problems. They comment that this weakness is particularly significant in the evaluation of direct manipulation software:

> Isolating problems may be a strength of this approach, but DM [direct manipulation] requires a more precise diagnosis than tends to be afforded. Interpretation of design faults is more complex, and it is this complexity which needs to be tackled in DM evaluation. (Springett & Grant, 1993, p. 252)

The validity of a user's intuitive assessment of a direct manipulation interface depends on the user correctly interpreting the relationship between direct manipulation actions and changes in the system state; a relationship which research indicates as being poorly understood (see, for example, Payne 1991, Sutcliffe & Springett 1992 and Springett 1992 for a discussion of this in the context of the use of *MacDraw*). In this sense, heuristic evaluation is prone to miss serious errors when applied in a direct manipulation context.

A notable feature of the research methodology adopted in this research is the way weaknesses (and strengths) of the various evaluation techniques are recognised. Most software evaluations feature the exclusive use of one evaluation technique. However, this research features three evaluation approaches, with the use of each technique matched to its inherent strengths. An initial heuristic evaluation of *Bioview* was conducted to identify important research issues from the user's point of view, a formal model of human-computer interaction in direct manipulation environments was developed to provide a model for the evaluation of direct manipulation educational software, and an extensive empirical study was conducted to identify flaws in the interface design of *Bioview*.

The heuristic evaluation was conducted during a preliminary phase in which a group of experienced educational software users were asked to use *Bioview* and pass comment on its design. In addition two members of this group subsequently conducted small scale empirical studies of the classroom use of *Bioview*. The results of the heuristic evaluation were used in two ways. Firstly, the issues emerging from the evaluation were used to inform the development of the model described in Chapter 2 of human-computer interaction. Secondly, specific issues which appeared worthy of in-depth research were identified. The results of the small scale studies were used to clarify these issues further. In addition the data collection and analysis problems encountered in one of these studies provided guidance on the choice of appropriate data collection and analysis techniques

In the major phase a detailed empirical evaluation of *Bioview* was conducted. Laboratory-based sessions of the use of *Bioview* with the PSYNTH datacube to answer set questions on how the rate of photosynthesis is affected by environmental factors were video recorded. These video records have been analysed in terms of the model of human-computer interaction developed in Chapter 2 with two aims in mind: (i) to critique the design of *Bioview* in particular, and (ii) to discuss the design of direct manipulation educational software in terms of learner and designer models.

The research subjects were interviewed after the completion of the laboratory sessions, and audio transcripts of these interviews have been transcribed. The transcripts have not been formally analysed, but they have been used to provide informal confirmation of the results of the analysis of the video records.

5.1 Preliminary research

The aims in the preliminary phase were to:

- indicate design and evaluation issues which users' perceived as important with respect to the use of *Bioview* specifically, and *Windows* based direct manipulation educational software generally
- indicate appropriate data collection techniques for the observation of human-computer interaction when students use direct manipulation software to complete educational tasks

- inform the development of an appropriate method to represent human-computer interaction when students use direct manipulation software to complete educational tasks
- identify conceptual issues associated with the tasks prescribed by the set questions
- determine the level of prior software expertise expected of *Bioview* users

A class of Masters students following a course on the design and evaluation of educational software provided a group of informed users capable of conducting a worthwhile heuristic evaluation of *Bioview*. This evaluation was guided by the completion of a questionnaire as described in Section 5.1.1. In addition to the questionnaire returns, data collected from this group included the transcript of an audio-taped group discussion, and extensive *Bioview* assessment reports completed by two members of this group as the course assessment item.

In preparation for the major data collection phase the set questions to be used as the basis of the laboratory sessions were analysed from conceptual and human-computer interaction perspectives. In addition, the textbooks that the observation subjects used were inspected to identify the approach to teaching limiting factors in photosynthesis that the subjects were familiar with, and the *Bioview* user documentation was analysed to ascertain the level of software expertise expected of users.

5.1.1 Clarification of research issues

The questionnaire that was administered to the group of Masters students is shown in Appendix 4. The questionnaire items were intentionally open-ended in an attempt to prompt a broad consideration of assessment issues:

- Please describe your reactions to the software as you familiarised yourself with the way the software operated.
- What curriculum areas do you think the software would be relevant to? Please explain your comments.
- Do you think the software would be accessible to the students you teach or you imagine being taught by other teachers? Please give reasons for your answer.
- What do you think the benefits of using this software would be? Please explain you comments.
- What do you think the potential benefits and weaknesses of the user interface design would be in the use of the software by pupils?

The students were also given the opportunity to make any other comments that they wished.

The responses to each questionnaire item were grouped together and inspected to identify common research issues. These issues were used to provide a framework for the analysis of the responses, with comments made in the responses matched to these research issues. This framework was also used to focus the inspection of the two assessment reports written by Masters students.

5.1.2 Observation techniques

Dix et al. (1993) identify five techniques for recording user actions: the completion of paper and pencil records during observation sessions, audio recording of comments made by users while they use software, video recording of the interaction between a user and a program, automatic logging by the system of user actions, and the completion of log books by users. The paper and pencil approach is limited by the researcher's writing speed, making it difficult to record detailed information. Audio-taping is most appropriate when a "think-aloud" protocol is used. However, it is usually very difficult, if not impossible, to specifically relate elements of the transcript to user actions, and it is difficult to match transcript elements to data recorded using some other protocol such as a written record of the interaction. Video recording has the great advantage of creating a record in which it is possible to see the actual actions executed by the user. The need to focus the camera on the user can inhibit the recording of the use of software in natural contexts, so this form of observation is best done in a laboratory setting. Automatic logging is usually relatively easy to arrange, but the "semantic free" nature of the data has already been alluded to. User logs rely heavily on the commitment of the user for them to be employed successfully. Typically the records will be made at a very coarse level - typically every hour rather than every few minutes.

Thompson (1991) described a variety of observation techniques as part of her report on the use of *Bioview*. These included audio-taping, the completion of user logs, and pencil and paper based observations (see Section 6.1.3.3). She was unable to match elements of the audio transcript to user actions and found that the user logs were not completed adequately to yield any useful information. The only method that she adopted which yielded results was the pencil and paper approach. She noted in the conclusion to her report that video recording would probably be the most effective form of observation.

5.1.3 Human-computer interaction representation

As described in Chapter 2 there are a variety of well known analytical models of human-computer interaction. Each model stresses particular aspects of human-computer interaction, typically focusing on system aspects or task aspects. This makes it possible to match these models to specific requirements. As explained in Chapter 2, the common established models are not ideally suited to representing direct manipulation. Of necessity the choice of one of the established models will involve the acceptance of a less than ideal way of representing direct manipulation.

However, an attempt was made to identify an existing model which could be adapted and modified for inclusion in a more extensive model specifically designed to address the features of direct manipulation environments. In order to do this, task related aspects of the use of *Bioview* were considered as a basis for identifying a model. The set questions were examined to formulate alternative "successful" ways of using the program to answer each question. The nature of these successful methods was then used to choose an appropriate analytical model.

The successful methods of using *Bioview* to answer each of the set questions could all be described in terms of goals and sub-goals. For example, in Question 1 the top-level goal is to find the optimum temperature for the maximum rate of photosynthesis. The question instructs the student to make sure that neither light nor carbon dioxide are in limited supplies. This leads to two sub-goals; set the light intensity at a non-limiting value and set the carbon dioxide concentration at a non-limiting value. The goal directed nature of the set questions implied that the GOMS model would be a possible candidate.

A more detailed inspection of sub-goals associated with each question is given in Section 6.2.3. This inspection revealed that each sub-goal could be represented by a unit-task consisting of the serial execution of a small number of well-defined operators. In addition, it was possible to identify three methods which could be applied in answering Questions 1, 2, and 3, and which could be used in combination to provide methods which could be used to answer Question 4 and 5. Each of these methods consisted of a small number of hierarchically linked unit-tasks. Thus the GOMS model was seen as an appropriate analytical model; the task in hand could be described in terms of goals and sub-goals, user actions could be described in terms of a limited set of well-defined operators, strategies for use could be expressed as well defined methods consisting of unit-tasks, and there was a range of methods to choose from.

5.1.4 Task related conceptual issues

The set questions provided a well-defined set of tasks which could be analysed prior to the observations of the use of *Bioview*. This analysis was conducted from a task perspective to identify the concepts implicit in the questions. The relevance of documented misconceptions in understanding the notion of interacting factors in general, and the limiting factors in photosynthesis in particular were considered. In addition, relevant sections of textbooks used by the observation subjects were studied to ascertain the pedagogical approaches to limiting factors in photosynthesis that they were familiar with.

5.1.5 Prior software expertise

As *Bioview* is a *Windows* application it is a reasonable assumption that users would be expected to be competent in a sub-set of *Windows* operations. In order to determine what this sub-set was the Users' Guide was inspected to establish the screen manipulation operations that the users were expected to be familiar with.

5.2 Laboratory-based observations

The laboratory based sessions were designed to provide an opportunity for the detailed observation of the use of *Bioview* with the PSYNTH datacube. The methodology described in this section was designed to enable the analysis of the extensive data collected during these sessions in terms of the task and system models held by the research subjects.

The experience of Thompson (1991), together with a consideration of the established strengths and weaknesses of the various observation techniques, was used to decide upon an appropriate observation technique. It was decided to make laboratory based video records of users interacting with *Bioview*. The laboratory location of the observations enabled a sophisticated video recording procedure. Two cameras were used; one camera was focused on the computer screen and the other camera was focused on the users. It was possible for the researcher to switch the recording channel between these two cameras, enabling both the interaction between the screen and the movements of the users to be recorded.

The failure of the user log approach reported by Thompson prompted a consideration of how to effectively record user's intentions. A means needed to be found by which users could be encouraged to articulate their intentions without being distracted from the task in hand. The approach taken was to ask one student to teach another student, with the expectation that the teacher-students' intentions could be inferred from their explanations and the teaching strategies that were adopted. The audio components of the video records were transcribed to provide a record of the these teaching sessions.

Three sessions were observed. In each session the teacher-student (Delia, Sharon, or Ruth) was an Upper 6th Form student currently following an A-level course in biology. They had been taught by the researcher on a previous occasion how to use *Bioview*, and they had had an opportunity to study the set worksheet questions prior to their participation in the laboratory sessions. The students taught by Delia and Sharon (Alice and Uri respectively) were both Lower 6th Form students. The student taught by Ruth was a Post-graduate Certificate in Education student.

Each session has been divided into a sequence of well defined task related episodes, each of which consisted of between one and four unit-tasks. Typically an episode corresponded to an attempt to answer one of the set questions, although each session also contained an episode corresponding to an illustration of the basic features of *Bioview*, and the final episode in one session was concerned with a general discussion of the merits of the program. A record for each episode has been completed which gives a statement of the top-level goal, and, if appropriate, the applicable successful method. Each record also includes comments on the observed human-computer interaction, and provides the following for each unit-task:

- the observed action substring
- an action substring corrected for software manipulation errors, such as the execution of a redundant con_graph operation
- an action substring reduced to take account of the execution of window manipulation and graph selection operations which were not integral to the applied method
- an "expert" action substring
- display and system "signatures" giving changes in the display and system states
- an annotated version of the transcript of the discussion

An example of an episode record is shown in Appendix 5. Episode records have been used to write full descriptions of each session which are included as Appendix 6.

The episode records have been used to create a two dimensional analysis of the human-computer interaction observed in the sessions. One dimension consists of an analysis in terms of the system use in each session, while the other dimension consists of an analysis in terms of each of the tasks defined in the set questions. These two analysis dimensions are considered in the following sections.

5.2.1 The analysis of system use

Analysis in terms of the system was completed by looking at the interaction history and by examining the implementation of the methods applied in each of the episodes. The methods of analysis which were applied to do this are illustrated with reference to a hypothetical Episode X.

5.2.1.1 The identification of methods

The annotated audio transcript and the associated action string that have been recorded in each episode record have been used to identify the method corresponding to each episode as an application of (i) a successful method, (ii) an idiosyncratic method, or (iii) recall of previous experience of using the program. To illustrate this process consider a hypothetical Episode X corresponding to an attempt to explore the effect of changing the level of carbon dioxide at the optimum temperature and the maximum light intensity. The observed action string for this episode is shown below. Redundant operations which had no effect on the state of the system or the display are enclosed in "curly" brackets:

|| s_sheet(T) | m_sheet(T) || s_row(C) | {con_graph(1)} | m_row(C) || s_graph(b)
| inspect_graph(r/L) || s_sheet(C) | m_sheet(C) | s_sheet(T) | repeat 2[m_row(C)
| inspect_graph(r/L) | scope_win]

An inspection of the transcript of the discussion between the teacher-student and the student indicates that the episode consisted of three unit-tasks: set the temperature at the optimum temperature, identify the rate of photosynthesis which corresponds the maximum light intensity, and explore the effect on the rate of photosynthesis of changing the value of the level of carbon dioxide. The action sub-strings which correspond to each of these unit-tasks are shown in Table 5.1.

Unit-task 1 was completed in an "expert" fashion; no errors are committed and the unit-task sub-goal directly matched the sequence of operations. In Unit-task 2, a redundant con_graph(1) operation was committed in error and an unnecessary m_row(C) was executed. An inspection of the audio transcript reveals the reason for this error. When the C-row was selected the default row position corresponded to the minimum C-value. This row consisted of a set of zero rate values and resulted in the display of a "null" rate/L graph, prompting the execution of a "panic" con_graph(1) operation to confirm that an operation had actually been executed. Execution of an m_row(C) operation resulted in a display of an instance of a rate/L graph. A bar chart was selected to make the identification of the maximum L-value easier.

Table 5.1: Action sub-strings corresponding to each unit-task in Episode X.

Unit-task	Description	Action sub-string
X/1	Optimise the temperature	ll s_sheet(T) l m_sheet(T)
X/2	Maximise the L-value	ll s_row(C) l {con_graph(1)} l m_row(C) l s_graph(b) l inspect_graph(r/L)
X/3	Vary the C-value	ll s_sheet(C) l m_sheet(C) l s_sheet(T) l repeat 2[m_row(C) l inspect_graph(r/L) l scope_win]

An expert application of Unit-task 3 would simply have consisted of repeated inspections of the rate/L graph as m_row(C) operations were executed. However, the observed action string for this unit-task indicated the existence of a misunderstanding of the appropriate way in which the C-value should be varied while keeping the temperature fixed at the optimum value. Initially the C-value was changed by executing an m_sheet(C) operation rather than a m-row(C) operation. A s_sheet(T) operation was executed to return to the optimum temperature situation. This misconception of the appropriate way to vary the C-value led to an initial "s_sheet(C) l m_sheet(C) l s_sheet(T)" sequence in the action sub-string for this unit-task.

An examination of Episode X illustrates that it is possible to associate a successful method with an episode even if the method is not executed in an expert fashion. Idiosyncratic methods correspond to episodes for which it is not possible to identify a successful method.

5.2.1.2 Examination of the implementation of successful methods

The observed action string for a successful method can be corrected for the commission of errors. For example, in Unit-task 2, the action sub-string would be corrected by deleting the redundant con_graph(1) operation. The observed action string would also be reduced by removing screen and graph manipulation operations that were not integral to the successful method - the s_graph(b) operation and repetitions of m_row(C) and scope_win operations necessitated by the need to change the scope of the datasheet window. The expert action sub-string for Unit-task 3 would be obtained by deleting the initial "s_sheet(C) l m_sheet(C) l s_sheet(T)" sequence. Table 5.2 shows the observed, corrected, reduced, and expert action sub-strings for the unit-tasks in Episode X.

Action string lengths for each unit-task can be determined for observed (n_o), corrected (n_c), reduced (n_r), and expert (n_e) sub-strings by counting the number of operations in each sub-string. The corresponding values for the whole episode are simply the sum of the values for the unit-tasks.

Table 5.2: Observed, corrected, reduced and expert action sub-strings for Episode X.

Sub-string	Unit-task X/1	Unit-task X/2	Unit-task X/3
Observed	ll s_sheet(T) l m_sheet(T)	ll s_row(C) l {con_graph(1)} l m_row(C) l s_graph(b) l inspect_graph(r/L)	ll s_sheet(C) l m_sheet(C) l s_sheet(T) l repeat 2 [m_row(C) l inspect_graph(r/L) l scope_win]
Corrected	ll s_sheet(T) l m_sheet(T)	ll s_row(C) l m_row(C) l s_graph(b) l inspect_graph(r/L)	ll s_sheet(C) l m_sheet(C) l s_sheet(T) l repeat 2 [m_row(C) l inspect_graph(r/L) l scope_win]
Reduced	ll s_sheet(T) l m_sheet(T)	ll s_row(C) l m_row(C) l inspect_graph(r/L)	ll s_sheet(C) l m_sheet(C) l s_sheet(T) l m_row(C) l inspect_graph(r/L)
Expert	ll s_sheet(T) l m_sheet(T)	ll s_row(C) l inspect_graph(r/L)	ll m_row(C) l inspect_graph(r/L)]

These action string lengths for Episode X are shown in Table 5.3. An inspection of the action string lengths for each episode has been used to type the episodes in each of the sessions. A non-zero difference between the observed action string length and the corrected action string length (n_o-n_c) has been used to identify those episodes in which manipulation errors have been committed. A non-zero difference between the reduced string length and the expert string length (n_r-n_e) has been used to identify methods in which there may be a misunderstanding of how to manipulate the display to make appropriate changes in variables. An inspection of the values of (n_o-n_c) and (n_r-n_e) for

Episode X would indicate that the execution of the successful method associated with this episode included the commission of errors and a possible misunderstanding of how to use the datacube appropriately.

Table 5.3: Action string lengths for Episode X

Length	n_o	n_c	n_r	n_e	n_o-n_c	n_r-n_e
Unit-task X/1	2	2	2	2	0	0
Unit-task X/2	5	4	3	2	1	1
Unit-task X/3	9	9	5	2	0	3
Episode	16	15	10	6	1	4

The application of a truncated method is evident when the action sub-string length is less than that length determined for a full implementation of the episode. In Episode X, display related truncation could have resulted from the T-sheet being selected at the end of the previous episode, making the execution of an s_sheet(T) operation unnecessary. The action string lengths for this truncated version of the successful method applied in Episode X would then be as shown in Table 5.4.

Table 5.4: Action string lengths for Episode X (display related truncation)

Length	n_o	n_c	n_r	n_e	n_o-n_c	n_r-n_e
Unit-task X/1	1	1	1	1	0	0
Unit-task X/2	5	4	3	2	1	1
Unit-task X/3	9	9	5	2	0	3
Episode	15	14	9	5	1	4

Task related truncation would also result in a reduction in the length of the action string length for the episode. For example, if no attention was paid to the value of the light intensity variable when the effect of changing the carbon dioxide level was explored the inspect_graph(r/L) operation would be unnecessary and Unit-task 2 would be unnecessary. The action string lengths for this truncated version of the successful method applied in Episode X would then be as shown in Table 5.5.

Table 5.5: Action string lengths for Episode X (task related truncation)

Length	n_o	n_c	n_r	n_e	n_o-n_c	n_r-n_e
Unit-task X/1	2	2	2	2	0	0
Unit-task X/2	-	-	-	-	-	-
Unit-task X/3	9	9	5	2	0	3
Episode	11	11	7	4	0	3

The existence of both task and display related truncation would result in a combined reduction of the action string lengths as shown in Table 5.6.

Table 5.6: Action string lengths for Episode X (combined task and display truncation)

Length	n_o	n_c	n_r	n_e	n_o-n_c	n_r-n_e
Unit-task X/1	1	1	1	1	0	0
Unit-task X/2	-	-	-	-	-	-
Unit-task X/3	9	9	5	2	0	3
Episode	10	10	6	3	0	3

The discussion of truncated versions of the method applied in Episode X indicates that even when the application of a method is truncated due to the display, the task, or both, it is possible to identify the occurrence of errors and misconceptions about datacube manipulation.

5.2.1.3 Examination of system register changes

The state of the system at the completion of each unit-task has been defined in terms of the system register; a surrogate model which mimics the operation of the *Bioview* core language in terms of a higher level representation. This register consists of three sub-registers, each of which corresponds to one of the three commonly orthogonal datasheets. Thus for the PSYNTH datacube there is an L-sheet sub-register, a C-sheet sub-register, and a T-sheet sub-register. Each sub-register is composed of three elements which give the values of the the three variables associated with the appropriate sheet. In the case of the

PSYNTH datacube these will be the L-value, the C-value and the T-value. Each element can take a maximum value (coded as 3), an intermediate value (coded as 2), or a minimum value (coded as 1). The sub-register corresponding to the currently selected data sheet is referred to as the active sub-register, as it is only values in this sub-register that can be changed unless an s_sheet operation is executed to change the type of datasheet that is selected.

The way in which the system register and the display register define the state of the software environment is illustrated by considering Table 5.7 which shows the system register at the completion of Unit-tasks 1, 2, and 3 in Episode X.

Table 5.7: System registers for the unit-tasks in Episode X

Unit-task	System register								
	L-sheet			C-sheet			T-sheet		
	L	C	T	L	C	T	L	C	T
X/1	1	1	1	2	1	1	2	1	2
X/2	1	1	1	2	1	1	2	2	2
X/3	1	1	1	2	2	1	2	3	2

As a T-sheet was selected at the end of Unit-task 1 the T-sheet sub-register will be active and only values in this sub-register could be changed unless an s_sheet(L) or s_sheet(C) operation is executed. An inspection of the active system sub-register confirms that the T-sheet was located in a position corresponding to an intermediate T-value (shown as 2) and that the highlighted L-column was located in a position corresponding to an intermediate L-value (shown as 2). The C-value was a minimum (shown as 1), indicating that if an s_row(C) operation was executed the highlighted row would be located in the position corresponding to the minimum C-value. The values in the L-sheet and C-sheet sub-registers show the sheet, row and column settings which existed at the start of the episode. The L-sheet sub-register shows sheet, row and column locations corresponding to minimum variable values, while the C-sheet sub-register shows the C-value at a minimum value, the L-value at an intermediate value, and the T-value at a minimum value.

At the end of Unit-task 2 the C-value in the active T-sheet sub-register had changed from the minimum value to an intermediate value, that is from 1 to 2. This was a consequence of the unnecessary execution of the m_row(C) operation in Unit-task 2. No changes in the L-sheet or C-sheet sub-registers were evident, indicating that no operations had been executed on the C-sheet or the L-sheet during this unit-task. (This may not be the case as a sequence of operations followed by a reverse sequence could have been executed to leave the active system sub-register in an unchanged state).

Although the T-sheet sub-register was active at the beginning and end of Unit-task 3, an inspection of the change in the C-value element from 1 to 2 in the C-sheet sub-register reveals that operations had been executed on the C-sheet during this unit-task. This change corresponded to the incorrect "s_sheet(C) | m_sheet(C) | s_sheet(T)" sequence of operations at the start of the unit-task. The execution of the m_row(C) operations left the C-value for the T-sheet at a maximum, as shown by an element value of 3 for this variable.

The interaction histories shown in Appendix 7 chronicle the state of the system register at the end of each unit-task as *Bioview* was used in each of the laboratory sessions. Episodes that involved changes in the active system sub-register have been identified. When a different sub-register was active at the start and end of an episode the episode has been coded as a "Δ" episode. When there have been changes in active system sub-register, either within a unit-task or within an episode, which do not result in a net change in the active system sub-register the episode has been coded as "∂" episode. The full implementation of the successful method in Episode X provides an example of a Δ episode; a s-sheet(T) operation was executed at the start of the episode indicating that a different sub-register (either a L-sheet or C-sheet sub-register) was active at the end of the preceding episode. The example of a display-truncated version of this successful method provides an illustration of a ∂ method; although the T-sheet sub-register is active at the start and finish of the episode, the C-sheet sub-register was also active during the episode.

5.2.1.4 Examination of display register changes

The state of the display at the completion of each unit-task has been defined in terms of the display register. This register consists of a sub-register for the datacube display and sub-registers for each of the graph_windows displayed on the screen. The datacube sub-register defines which datasheet was selected, and whether a row or column is highlighted, upon completion of a unit-task. A graph window sub-register defines attributes of the window upon completion of a unit-task: (i) whether the window was connected or not, (ii) what instance of rate/variable graph was displayed, and (iii) which of a bar graph, line graph, or pie chart is shown.

The datacube display register at the end of each unit-task in Episode X is shown in Table 5.8, and the graph window display registers are shown in Table 5.9. The only active graph window during this Episode X was graph_window(1). The instance of the graph shown in graph-window(2) corresponded to a previous episode. An inspection of the display sub-register states for each unit-task that are chronicled in Appendix 7, enables a review of the changes in both the datacube display and the instances of graphs shown in the graph_windows.

Table 5.8: Datacube display register for Episode X.

Unit-task	Display register: datacube					
	L-sheet		C-sheet		T-sheet	
	row	col	row	col	row	col
X/1						•
X/2					•	
X/3					•	

Table 5.9: Graph_window display register for Episode X.

Unit-task	Display register: graph-windows											
	con		r/L		r/C		r/T		1		b	
	w1	w2	w1	w2	w1	w2	w1	w2	w1	w2	w1	w2
X/1	•				•			•	•	•		
X/2	•		•					•		•	•	
X/3	•		•					•		•	•	

5.2.2 Task related analysis

Each of the set questions is based on a clearly stated task, and, as such, these questions provide a cross-session task based framework for the analysis of the human-computer interaction observed during the laboratory sessions. An inspection of the record sheet for each episode concerned with one of the set questions has enabled method selections to be grouped according to each question. Based on the method selections in each group it is possible to consider the following questions for the task related to each set question:

- Do the students use the same or similar methods to the inferred successful methods?
- To what extent do the chosen methods use direct manipulation interaction to either fix the value of a variable or explore the effect of changing the value of a variable?
- How frequent is task based truncation of successful methods?

5.2.2.1 Comparison of inferred designer and preferred learner models

It is possible to identify successful methods which are applicable to each of the tasks stated in the set questions (see Section 6.2.3). A comparison of the set of successful methods for a given question with the methods actually selected by learners has been conducted to ascertain the correlation between inferred designer models and learner models of the software environment.

The existence of task related truncation indicates that the learner was engaged in the application of a method which involved the consideration of only one or two of the interacting variables. This may have been a conscious decision or may have arisen through a lack of appreciation of the importance of the interaction between the variables. Learner method selections corresponding to each set question have been inspected to identify the extent to which task related method truncation took place. This has provided an indication of the extent to which the students considered the interactions between the variables.

5.2.2.2 Direct manipulation of variable values

As discussed in Chapter 4, Method 1 involves the use of direct manipulation to fix the values of two variables by executing a combination of an m_sheet and an m_row/col operations. The exploration of how the rate varies with the third variable does not involve direct manipulation; it only requires the inspection of an instance of a graph of the rate plotted against the third variable. As Method 4 is a combination of two versions of Method 1 it also does not involve direct manipulation to vary variable values during the interpretation phase of the method. However, Methods 2 and 3 do involve the use of direct manipulation of a variable to vary a value through the execution respectively of an m_sheet or an m_row/col operation. By implication Methods 5, 6, 7, 8 and 9, which all include at least one application of Method 2 or Method 3 as a component, involve the use of direct manipulation to vary variable values.

An inspection of the method selections in each of the question based groups has been conducted to identify the prevalence of the use of direct manipulation for (i) fixing the value of a variable, and (ii) varying the value of a variable. This has enabled the use of direct manipulation to be considered with respect to well-defined task constraints and requirements.

5.2.3 Applicability of the GOMS model

The applicability of the GOMS model to the analysis of the human-computer observed in the laboratory sessions was assessed by considering separately each of the four components of the model: goals, operators, methods, and selection.

5.2.3.1 Goal structure and definition of operators

If the GOMS model is to be applicable it should be possible to describe the task related to an episode in terms of a top-level goal and a small number of associated sub-goals. Each sub-goal should relate to a unit-task consisting of the serial execution of a limited number of well defined operators. In order to assess whether this goal based description was applicable to the laboratory sessions the following analyses were conducted:

- The annotated transcripts corresponding to each of the sessions that featured a successful method were examined to ascertain whether the interaction in the episode was goal based. In addition, the initial illustrative episodes in each session and the long confused Episode R6, which featured a number of idiosyncratic methods, were examined to see if they matched a goal based structure.
- The number of unit-tasks per episode was counted for each successful method to indicate the typical number of unit-tasks per episode.
- The sufficiency of the set of operators identified in Chapter 4 was assessed.
- Reduced action string length for each successful method were calculated.

5.2.3.2 Method identification and selection

If the application of well defined methods is a feature of the successful use of *Bioview*, it should be possible to allocate a well-defined method to each episode which featured a successful attempt to answer one of the set questions. A review of the number of task related episodes which were successfully concluded was completed with the aim of identifying the number of these episodes in which the human-computer interaction could be represented in terms of a successful method. In addition a number of selection rules for each of the set questions was formulated in terms of the influence of display related method truncation. The number of times that each of these rules appeared to be used was determined for each of the episodes.

5.3 Summary

The research programme featured a preliminary phase based on an heuristic evaluation of *Bioview*, a consideration of the results of two small scale empirical studies, and an analysis of related teaching materials. The principal phase of the research consisted of an in-depth empirical evaluation of three laboratory sessions in which the use of *Bioview* was video recorded. The findings from the preliminary phase were used to inform both the development of a formal model for the use of direct manipulation educational software and to clarify research issues and an appropriate methodology. The empirical evaluation featured analyses of (i) human-computer interaction from a system perspective, (ii) human-computer interaction from a task perspective, and (ii) the applicability of the GOMS approach for representing human-computer interaction in direct manipulation environments.

Chapter 6

Preliminary data analysis

In this chapter data collected during the preliminary data collection phase are analysed. Data were collected in this phase from three sources - the heuristic evaluation of *Bioview* by a group of Masters students, the two small scale empirical evaluations of the classroom use of *Bioview* by two of these Masters students, and an inspection of educational and user documentation relevant to the use of *Bioview*.

6.1 Heuristic evaluation of *Bioview*

Bioview was briefly presented by the researcher to the group of 14 Masters students, none of whom had seen or used the program before. The students were asked to evaluate the program with the aid of the questionnaire shown in Appendix 4. The students spent approximately one hour on this evaluation exercise. A week later the group of students and the researcher discussed their evaluations of *Bioview*. The discussion was audio taped and transcribed.

6.1.1 Questionnaire response analysis

The responses to each questionnaire item were grouped together. Each group of responses was inspected to identify assessment issues. The *Windows* environment, the datacube metaphor, graph display features, and the use of dynamically presented instances of graphs emerged as assessment issues. The responses were then re-grouped according to these four issues and inspected to identify specific concerns.

6.1.1.1 The *Windows* interface

An inability to use the *Windows* interface was perceived as an initial barrier to the effective use of the software:

> Certain pre-requisites [are] needed before [the] software can be fully used - *Windows* mouse pointer etc. (Student 1)

> Unlikely to discover range of potential for WIMP use without help. (Student 5)

> The class should be reasonably computer literate or the mechanics will operate as a bar to the data handling. (Student 7)

> Without experience of a "*Windows* type" interface I feel the students would find it difficult to cope without very much clearer instructions. (Student 9)

> It would seem that prior knowledge [or] experience of *Windows* would be advisable. (Student 12)

Although other students thought that mastering the *Windows* interface would provide initial problems they also felt that these problems could be overcome relatively easily. Student 3 illustrates this perception:

> Once I mastered - in a small way - *Windows* I quite quickly began to see what the software was allowing me to do - this then allowed to focus quite rapidly (and quicker than I expected) on an exploration of the data. (Student 3)

The context provided by *Bioview* was thought by some students to be an aid to coping with *Windows*:

> Students would need to competent in the use of *Windows*. The notion of a cube as a model to represent data is novel and helps to make it a more concrete experience. (Student 8)

> If students are familiar with the *Windows* environment then starting with simple examples [in *Bioview*] the software would be quite accessible. (Student 13)

Problems concerned with the interpretation of multiple overlapping windows were noted. Student 1 commented that there was "lots happening on the screen", Student 2 felt that it was "easy to get a messy screen", and Student 12 thought that the possibility of the screen looking "cluttered and untidy" could be off-putting. The comments by Students 11 and 12 emphasised this point

> You had to think carefully about what is happening on the screen It was a bit overpowering at first. (Student 11)

> Very slow in getting started. Took some time to view four (or just two) windows at the same time. The spreadsheet display was very limiting, and enlarging it covered up too much of the other windows, particularly the *Bioview* and graph windows. Gradually got used to it, however. (Student 12)

A number of students commented on the need for design consistency with other *Windows* applications. The belief that this led to the transfer of skills between the use of packages was evident in the comment of Student 3:

> Yes, the students are increasingly becoming familiar with DTP packages which use [the] same principles and thus the accessibility would not be a problem for them. (Student 3)

The strength of this belief was clear from the criticism of by Student 6 of the non-adherence of the design of *Bioview* to *Windows* guidelines:

> [...] it simply doesn't look like a proper *Windows* application. [...] So why doesn't the file menu look like a proper file menu with New at the top and Print, Printer Select, Exit, and About *Bioview* ? Where did Edit menu go? Diminish and Enlarge should be under View. Tidy should be replaced by a Window menu. (Student 6)

6.1.1.2 The datacube representation

Some students found it initially difficult to understand the datacube representation. Student 1 thought that "time was needed to visualise what is happening when clicking on arrow for species, year, sample", and Student 7 found that it "takes a little time to get used to the correlation of the numbers to the changing shape of the cube". The graphical form of the datacube caused some concern. Student 5 was confused over the purpose of the dotted line displayed on the datacube, and Student 11 felt that "the graphics should give the student a clearer impression of the learning objective". However, other students found the representation helpful:

> [...] The notion of a "cube" as a model to represent data is novel and helps to make it a more concrete experience. (Student 8)

> It gives an overall view of complex data that would not be possible with printed tables. It extends the notion of spreadsheet to three dimensions. (Student 13)

The mixture of two-dimensional and three-dimensional representations involved in the datacube was commented on:

> The cube metaphor can be misinterpreted due to [...] use of two planes of description for three planes of use. (Student 5)

> Pity that the 3D cube is only able to produce 2D sheets - why not transparent cubes? (Student 2)

6.1.1.3 The use of graphs

Some students experienced difficulty in launching graph windows. Student 1 found the "graph word [that is, menu item] hard to find". Student 7 experienced difficulty in finding the graph icon and found it hard to use the scale option.

Student 7 found the "instant changes in graphing impressive" and "graph creation and importing very fast and motivating". This impression was echoed by Student 5:

> Turning data into a dynamic representation of reality. To provide a sort of "data animation" which may lead to a more meaningful perception of data tables and the events they record. (Student 5)

However, Student 9 felt that "a good deal of practice by both teacher and student would be necessary to familiarise them with interpreting "dynamically" represented software, and Student 2 felt that it became "too easy to produce graphics without thinking what they mean"

6.1.2 The group discussion with Masters students

The 14 Masters students divided into three discussion groups and a nominated member of each group reported on their discussion. Each group thought that the restricted application of *Bioview* to biology failed to realise the potential of the software, and a wide range of cross curricula activities were envisaged.

Most of the feedback was concerned with comments on the role of *Windows* in the design and use of the software. All groups reported a concern that *Windows* could act as a barrier to the initial use of *Bioview*. A comment made during the feedback from the first group illustrated this concern:

> So really there was an issue about initial familiarity and familiarisation, which had perhaps more to do with the use of *Windows* itself; which is a sort of a model of a desk-top, and having access to a variety of tools, and being able to push some out of the way, and use the others, and prioritise the things as you saw fit. And once you got the knack of that I think it made *Bioview* as a tool easier to use.

A "wish-list" of improvements to *Bioview* included the suggestion that a software utility should be written to enable the import of data from other database packages. This utility would be able to extract the data values of a dependent variable that corresponded to the values of three independent variables. Another item in this list concerned the form of the graphical displays used by *Bioview*. The suggestion was made that the result of an m_sheet operation should be recorded in order to produce a three dimensional graph representing the changing instances of the two dimensional graph which was displayed as the m_sheet operation was executed.

6.1.3 Small scale empirical studies

Two students (Barbara Thompson and Tony Hill) elected to write a 10 000 word report on the school-based assessment of *Bioview*, which were made available to the researcher for analysis. The report by Thompson reported varied and fairly extensive observations of the use of *Bioview*. Her report included a personal preliminary assessment, reviews of the software by two science teachers, and observations of the use of the software by two groups of Year 9 students following a course in general science and two groups of A-Level biology students. Hill reported observations of the use of *Bioview* by two Japanese students studying the equivalent of A-Level biology and a science teacher group.

6.1.3.1 Preliminary assessment of *Bioview*

Thompson described herself as "not really familiar with *Windows*" in one of her responses to the questionnaire completed by the Masters students. In her report she said that she was preoccupied with how to operate *Windows* during this preliminary assessment rather than the educational use of *Bioview*. This supports the notion that the operation of *Windows* may act as a barrier to novice users. However, she successfully worked through the Users' Guide until she came to the instructions to create a second graph window on the screen when she was unable to find any information on how to start a second graph. She described her experiences as follows:

> The first unconnected graph I had drawn was produced by double clicking on the graph icon and then pointing to the Connect label to graph the data. Now there was no graph icon to click on and I found that to get a second graph at this stage was impossible and I was then reduced to trial and error activities [...] This difficulty remained unresolved until one of the biology teachers in the school, after spending at least 40 minutes on the program, and going through all the options on the pull down menus, realised that the Start menu had the word graph on it and this was what allowed the user to create further graphs. (Thompson,1991, p. 11)

Thompson (1991) commented that "the more I used the package, the more I made of the datacube window for navigating my way round the data' (p. 13). She also noted the possible use of the View option:

> It is fair to say that in spite of this "model" of data in the form of a cube the software can be readily understood by using the View menu option which allows an alternative way of looking at the planes of data. (Thompson, 1991, p. 13)

The possible conflict between three dimensional and two dimensional representations was perceived as a possible source of confusion:

> In order to try and portray the concept of a three dimensional database the software employs a model of a cube in which the three different planes of the cube represent the three variables under consideration. The three variables can all change at the same time so it is possible to see how they interact. However, only two variables can be considered at any one time. This may be confusing for the user inasmuch that any one point on the cube represents data related to the three interacting variables but the display of the data on the datasheet or the graph is two dimensional and involves only two variables. (Thompson, 1991, p. 12)

This perception of how the variables can be changed using the datacube indicated a misconception of how manipulation of the datacube related to changes in the system register; although it was possible to change the value of any one of the variables at a given time they could not all be changed at the same time as stated by Thompson. If her interpretation of how to manipulate the datacube had been correct it would have been possible to simultaneously change all the values in a system sub-register, whereas it is only possible to change one value at a time. The assertion that only two variables can be considered at any one time implies that the notion of the datacube consisting of a series of datasheets was being used by Thompson. Her feeling that this might cause confusion due to a two dimensional representation of three interacting variables again indicated a misconception of the relationship between the datacube structure and the system sub-register; the possibility of altering the value of the third variable in the active system sub-register by moving the location of the selected datasheet was not considered. Note also that there was an apparent confusion between the two variables which correspond to a datasheet and a displayed graph.

6.1.3.2 Assessment by subject specialists

Two biology teachers reviewed the package. One teacher was an experienced *Windows* user and found no difficulty in using *Bioview*, apart from starting a second graph window. This teacher used the TRANSP datacube which is concerned with the effects of humidity, air speed, and temperature on the rate of water loss form a leaf. He was able to illustrate changes in the rate of water loss by using dynamically represented instances of graphs. The other teacher was a novice *Windows* user, but was able to operate *Bioview* quite easily after a short demonstration by Thompson and 15 minutes spent exploring the features of the software, which confirmed the expectations expressed by some of the Masters students when they completed the questionnaire. This teacher is interesting because she felt motivated to use the PSYNTH datacube after her introduction to *Bioview*:

> That particular day [the day she was introduced to *Bioview*] she had an A-level photosynthesis experiment set up in the laboratory which failed to work. The experiment was meant to demonstrate the limiting factor effect of carbon dioxide

> concentration in photosynthesis using the pond weed Elodea. She therefore wanted to recreate the experiment using *Bioview* whereby she could gradually increase the concentration of carbon dioxide. She was eventually able to create the necessary graph which enabled her to step through the carbon dioxide levels and thus be able to see from the dynamic representation where the limiting factor phenomena came into play. (Thompson, 1991, p. 26)

6.1.3.3 Analysis of pupil use

The groups observed are shown in Table 6.1. All the groups used *Bioview* with the DISEASE datacube, which gives the annual incidence of death in various parts of the world from four infectious diseases. Each group was asked to produce an ordered series of graphs on the screen to show the incidence of malaria across the world in 1975, 1976 and 1979. Before they attempted this task Thompson introduced each group to the following features of *Windows*: (i) the idea that *Windows* is a based on a desk-top metaphor, (ii) multitasking; (iii) loading and saving files; (iv) maximising and minimising windows; (v) resizing and positioning windows; and (vi) connecting graphs.

Table 6.1: Student groups observed by Thompson

Group	Level	Subject	No.	Duration
1	Year 12	A-level biology	2	Two sessions of 1.75 hours
2	Year 12	A-level biology	3	Two sessions of 1.75 hours
3	Year 9	General science	3	One session of 1 hour
4	Year 9	General science	2	One session of 1 hour

Four data collection methods were used simultaneously during the observation of these groups. The sessions were audio-taped to record the group discussion as the students used *Bioview*. Initially the students were intimidated by the presence of the tape recorder, but they overcame this after about half an hour. However, Thompson found that the information on the tape was very difficult to interpret as there were long pauses between comments whilst they waited for the computer to respond, and most of the comments were very short and were mostly meaningless when heard out of context, that is, without a corresponding visual record of the screen display. As Thompson comments "there was little interpretation which could be put to such remarks as 'try clicking on that'" (Thompson, 1991, p. 32).

The students were asked to complete a log of problems, remedial actions, and the reason for these actions. This request did not result in any worthwhile information.

Thompson felt that the students lacked the "sophisticated style" necessary to express the rationales for their various actions, and that the pressure of the work in hand mitigated very strongly against students giving the attention required to meaningfully complete the logs. Thompson also asked the students to write a short assessment of *Bioview* using the headings of "what I have learnt", "what I liked about *Bioview*", "what I disliked about the program". There is no discussion in the report of these written assessments, and it can only be assumed that either the students did not complete them, or that they did not yield useful data.

The fourth approach was unstructured observation by Thompson in which she "simply recorded what they did and occasionally made a suggestion in order to allow them to progress" (Thompson, 1991, p. 51). This approach proved to be the only effective method of data collection, although Thompson noted that a more structured approach, possibly based on the use of a checklist, may have been more effective.

Group 1 only seemed concerned with operating the package, rather than considering the task in hand, during the first 15 minutes of the observed session. However, the longer they used the package the more adept they became at using *Windows*. The display of an appropriate sequence of graphs caused considerable problems, such as getting graphs on top of one another, of "losing" the first graph that was plotted, connecting an unintended graph, and restoring the wrong graph. The group did not appear to appreciate the possibilities of displaying multiple graphs as they produced three separate printouts of maximised graphs. Thompson notes that when they became aware of the need to produce more that one graph on the screen at a time they were unable to do this even though she showed them the "start menu" and also how to store the graph an icon. Although the group used the datacube to navigate their way around the software they experienced great difficulty in getting the datasheet set up in the correct format. They were constantly connecting to a graph window and produced a graph with years on the horizontal axis when they wanted to display countries on this axis, indicating a lack of appreciation of the relationship between row/column selection and the instance of the graph displayed.

The students in Group 1 suggested a further task - the creation of a transpiration datacube based on experimental data that they had recently collected. The datacube that they created revealed that they misunderstood the nature of the datacube system - they specified rows corresponding to different environmental factors (light, humidity etc.) on the same datasheet that already showed rows that corresponded to different values of distance of the plant from the source of environmental change, for example a lamp.

Group 2 initially focused all their attention on the mechanics of operating the software. As with Group 1, this group started by trying to produce a number of maximised graphs. Thompson had to remind them of the need to create three graphs on the screen at same time. Again she had to show this group how to start and iconise graph windows. The group were aware of the importance of the scale that was adopted, with the need to use the same scale if proper comparisons were to be made. Thompson reports that the students "intuitively chose scale from the graphs menu to change it but I suspect that they were unaware of what the scale represented" (Thompson, 1991, p. 39). This group showed no sign of producing any type of dynamic representation of data they were manipulating. The performance of Group 2 was summarised by Thompson as follows:

> By trial and error they arranged the datasheet so that the names of the different countries appeared across the top of each of each column, they listed the years vertically down the datasheet and then they used the cuboid to click through the disease variable until the disease malaria appeared. They then highlighted the appropriate year row and connected it to the graph. From the printout it can be seen that they kept the scale constant across the three graphs but the scale is too small. (Thompson, 1991, p. 40)

Practical considerations required Thompson to observe Group 3 and Group 4 at the same time. Group 3 adopted a trial and error approach and needed "to be reminded how to store the data in the form of icons" (Thompson, 1991, p. 43). Thompson reports:

> They "lost" several of their graphs after creating them, they were puzzled as to why this was happening. I eventually suggested that instead of storing the graphs they moved them around the screen so this entailed them removing the cuboid in order to have sufficient space. (Thompson, 1991, p. 44)

She also noted that the third girl found that there was too much on the screen for her to be able to determine exactly what was happening.

Although the group successfully completed the task there was no apparent mention of the intervening years (1977 and 1978), implying that the group did not move the disease/region datasheet through the datacube to display a dynamic representation of changes with time.

With some assistance from Thompson to start graphs and position them on the screen, Group 4 succeeded in displaying instances of disease/region graphs in three windows as requested. They very quickly started to generate graphs, selecting datasheets corresponding to different years. Thompson describes their behaviour in the following terms:

> [They] remembered how to store windows as icons [presumably from the initial demonstration by Thompson] and by trial and error sorted out the the order of the storage system. (Thompson, 1991, p. 45)

At an early stage the group realised the importance of having the same scale on each of the graphs. Achieving the same scale and producing readable graphs caused some problems:

> They realised that the larger the scale on their charts the clearer they were to read, they also wanted the same scale on the three graphs. They tried several times to do this unsuccessfully. (Thompson, 1991, p. 46)

> Not realising that because the numbers on their third graph were lower than the scale they were trying to use meant that they could not produce an easy to read graph. Eventually they did realise what the problem was and realised that they could not have three identical scales if they wanted clear, easy to read bar graphs. (Thompson, 1991, p. 46)

As the two students observed by Hill had not used *Windows* Hill gave a short demonstration of the interface, before leaving the students to explore the software for themselves. The exploration was based on the ENZYME1 datacube, which consists of rate of reaction values for six enzymes at various temperatures and pH values. Hill commented:

> Their inexperience coupled with the [...] freedom to roam around the desk-top resulted in chaos. With *Bioview* launched from its Program Manager (PM) icon, its tiled windows overlaid but did not remove the Program Manager that could be seen between the tiles. It had been inadvertently activated (thus becoming the top window). Having assumed that *Bioview* had now gone (it had simply been sent to the bottom of the metaphorical desk-top pile) frantic efforts to reload it from the now visible icon were underway. *Bioview* will neither reload another copy nor warn of one already available. [...] Once the the PM window was iconised (teacher led demonstration) [...] their navigation soon became more confident. (Hill, 1991, p. 15)

As with the Group 2, the students observed by Hill preferred to use the datasheet rather than the datacube. In answering the questions on the worksheet supplied with the documentation, they simply selected the View option, maximised the datasheet window, scanned the datasheet for the maximum value, recorded this value, and then used the View menu option to select another sheet corresponding to one of the other remaining enzymes, and repeated the process. Hill showed them how to investigate the variation of the maximum value with enzyme type by using the dynamic graphical representation linked to moving through the cube. However, the students preferred to stay with their method based on the use of the View option.

6.2 Analysis of related documentation

Relevant sections of the biology textbooks used at Wakeleys School have been inspected to identify the treatments of limiting factors in photosynthesis that the student-teachers were familiar with. The Users' Guide written to accompany *Bioview* (Riley, 1991) has been studied in order to establish the level of expertise in *Windows* and the knowledge of *Bioview* operations that was assumed by the designers. The worksheet questions in the Teachers' Guide were used in a slightly modified form as the basis for the laboratory observation sessions. These modified questions have been analysed to infer the design assumptions made about the learners' knowledge of interacting variables and limiting factors in photosynthesis, and to identify likely successful methods that learners would employ in answering the questions.

6.2.1 Biology textbooks used at Wakeleys School

The main text book used at Wakeleys School is *Biology for Advanced Level* (Toole & Toole, 1991). Two other textbooks are used for reference: *Biological Science 1 & 2* (Green, Stout & Taylor, 1990) and *Biology for Life* (Roberts, 1981).

All three books feature a discussion of the effects of light intensity, carbon dioxide concentration, and temperature on the rate of photosynthesis. Toole & Toole (1991) also discuss the effects of chlorophyll concentration, oxygen concentration, and the availability of water. Green et al (1990) also consider these additional factors, together with a brief discussion of the effects of pollution and specific inhibitors such as herbicides. Roberts (1981) only considers water as an additional limiting factor. None of the books includes a discussion of the notion of principal interacting factors.

Although a number of factors which affect the rate of photosynthesis are discussed in each book the discussion of limiting factors is restricted in all three books to a consideration of the interaction between light intensity and carbon dioxide concentration. Each discussion features a definition of the law of limiting factors which provides the basis for the discussions:

> At a given moment, the rate of a physiological process is limited by one factor which is in the shortest supply, and by that factor alone. (Toole & Toole, 1991, p. 324)

> When a chemical process is affected by more than one factor its rate is limited by that factor which is nearest its minimum value: it is that factor which directly affects a process if its quantity is changed. (Green et al , 1990, p 268)

> The rate of photosynthesis is controlled by several factors . At a particular moment the rate is determined by whichever factor is closest to its minimum value. (Roberts, 1981, p. 192)

This law is discussed by Tool & Toole with reference to a graph of rate of photosynthesis plotted against light intensity which shows three curves corresponding to (i) a low concentration of carbon dioxide and a low temperature, (ii) a higher concentration of carbon dioxide and a low temperature, and (iii) a higher concentration of carbon dioxide and a higher temperature. Toole & Toole explain the role of limiting factors in determining the rate of photosynthesis at a given point in the following way:

> If the amount of light given to a plant is increased the rate of photosynthesis increases up to point and then falls off. At this point some other factor, such as the concentration of carbon dioxide, is in short supply and so limits the rate. An increase in carbon dioxide concentration again increases the amount of photosynthesis until some other factor, e.g. temperature limits the process. (Toole & Toole , 1991, p. 325)

Green et al. (1990) state that:

> [...] it has been shown that different factors, such as carbon dioxide concentration and light intensity, interact and can be limiting at the same time, although one is often the major factor. (Green et al., 1990, p. 268)

They go on to consider the significance of the shape of a stylised graph of rate of photosynthesis against light intensity which shows three curves that correspond to (i) a temperature of 15 ^{o}C and the typical atmospheric concentration of carbon dioxide, (ii) a temperature of 25 ^{o}C and relatively high concentration of carbon dioxide, and (iii) 15 ^{o}C and a relatively high concentration of carbon dioxide *or* 25 ^{o}C and the typical atmospheric concentration of carbon dioxide. Referring to this graph Green et al. state that:

> Enzyme-controlled reactions like the dark reactions of photosynthesis are sensitive to temperature; thus an increase in temperature from 15 ^{o}C to 25 ^{o}C results in an increased rate of photosynthesis [...] providing light is not the limiting factor. Carbon dioxide concentration can also be a limiting factor in the dark reactions. [referring to the curve corresponding to a temperature of 15 ^{o}C and the typical atmospheric concentration of carbon dioxide] Thus [...] both temperature and carbon dioxide are limiting, and an increase in either results in an increased photosynthetic rate. (Green et al., 1990, p. 269).

Roberts (1981) refers to a graph of rate of photosynthesis against light intensity which shows two curves (A and B). Curve B corresponds to a higher concentration of carbon dioxide. The limiting behaviour of carbon dioxide is described as follows:

> First look at Curve A. Notice that as the light intensity is gradually raised, the curve rises, i. e. the rate of photosynthesis increases.
>
> However, there comes a point when the curve flattens out - in other words the rate of photosynthesis does not increase anymore, however much the light intensity is raised. Why do you think the rate of light intensity stops increasing? The answer is that some other factor than light is preventing photosynthesis from going any faster. We say that this factor is now limiting the rate of photosynthesis.
>
> What might this factor be? Well, it could be carbon dioxide. How could we find out if it is carbon dioxide? One way would be to raise the amount of carbon dioxide in the atmosphere surrounding the plant and repeat the experiment.
>
> The result of doing this is shown in curve B. This time a much higher rate of photosynthesis is achieved. What does this tell us? It tells us that carbon dioxide must have been limiting the rate of photosynthesis when the curve flattened out in the first experiment. (Roberts, 1981, p. 192)

Each book deals with the concept of optimum temperature. Green et al. simply state that the for temperate plants the optimum temperature is usually about 25 °C and indicate that rate doubles for every increase of 10 °C. Roberts reports the same rate of increase and notes that:

> Up to a certain point, the higher the temperature, the faster a plant will photosynthesise. (Roberts, 1981, p. 191)

Toole & Toole also report a doubling of rate for each increase of 10 °C up to the optimum temperature. They then state that:

> Above the optimum temperature, the rate of increase is reduced until a point is reached above which there is no increase in photosynthesis. The optimum photosynthetic rate for C3 plants is around 25 °C and for C4 plants lies between 35-40 °C. Above these levels further temperature increases lead to a levelling off and then a fall in the rate of photosynthesis. (Toole & Toole, 1991, p. 326)

6.2.2 *Bioview* Users' Guide

The Users' Guide (Riley, 1991) provides instructions on how to operate *Bioview*. A measure of the knowledge of *Windows* that users are expected to have can be inferred from the outline of *Windows* that is given in the Introduction. It is clear that users are expected to be familiar with the multi-tasking features of *Windows*::

> *Bioview* is able to provide several connected views of data on screen because it takes advantage of Microsoft *Windows*. This provides a framework in which several programs can be running at the same time and communicate with each other.
>
> Earlier software was restricted to one program running at a time and the program would take up the whole screen. With *Bioview* and other *Windows* programs, known as applications, you may have two, three, or more applications on the screen at one time. Each application has its own window(s) and you can choose where to place them on screen, how large they are and which ones overlap with the others. (Riley, 1991, p. 4)

Users are expected to be aware of the use of the clipboard:

> You can cut and paste data and graphs from *Bioview* into other *Windows* programs. For example, you may wish to do this to write a report in the *Windows Write* word-processor application and copy tables of figures and graphs from *Bioview* into your word-processed document. (Riley, 1991, p. 4)

The extent to which users are expected to manipulate the screen display is indicated by Riley in the descriptions he gives on how to configure and change the screen display:

Expand an icon into a window:
• point to an icon of a window (icons are usually 'parked' at the bottom of the screen)
• double-click on the mouse button and the window appears on the screen
or
• click once on the icon, so that the system menu appears
• point and click on Maximise in the system menu, and the icon expands into a window filling the screen

Reduce a window to an icon:
• point and click on the system button on the left of a window's title bar
• point and click on Minimise
or
• point and click on the single, downward-pointing arrow in the top right hand corner of the window

Reduce a full-screen window to a smaller window:
• point and click on the double arrow in the top right hand corner of the window

Change the position of a window:
• move a window to a new position by pointing to the title bar and dragging it to a different place.

Change the size of a window:
• point to one of the corners or sides of the window so the pointer on the screen becomes a double-headed arrow
• drag the window frame to a new position and release the mouse button.

To close a window (finish using that particular program)
• point and click on the system button to the left of the title bar
• point and click on Close when the system menu appears

Instructions on how to use *Bioview* itself are given in the form of tutorials; one tutorial is concerned with displaying data and the other tutorial is concerned with designing new databases and data entry. An inspection of the data display tutorial indicates the application specific operations that users are expected to be able to execute when they use *Bioview*. The tutorial provides instructions on how to execute s-sheet and m-sheet operations, and how to execute s_row/col and m_row/col operations. Instructions are also given on how to connect a graph window, execute an s_graph operation, and scale a graph. A curious omission is the lack of instructions on how to start new graphs, although the display of more than one graph window is considered. Cut and paste options are also dealt with in the tutorial.

6.2.3 Set worksheet questions

The questions posed in the Teachers' Guide were used as the basis for preparing a simplified set of questions to be used during the laboratory sessions. Some redundancy in the questions set in the Teachers' Guide was removed by reducing the number of questions from six to five. The simplification was felt necessary in order to make the time required to complete the questions consistent with the observation and analysis of the laboratory sessions.

The five questions that were used in the laboratory sessions were:

1. Use the data to find out the optimum air temperature for the maximum rate of photosynthesis. Make sure neither light nor carbon dioxide are in limited supplies.

2. At the optimum temperature look at the way in which varying the light or the carbon dioxide levels affects the rate of photosynthesis.

3. Try to find an optimum level of either light or carbon dioxide.

4. Is there any difference in the effect of increasing levels of light or carbon dioxide on the rate of photosynthesis?

5. At the fastest rate of photosynthesis shown in the data which factor would you try to increase more to attempt to increase the rate further?

These questions are considered in the following sections in order to infer the design assumptions which were made about how the software would be manipulated, and the assumptions that were made about learners' understanding of how environmental factors affect the rate of photosynthesis. In the following descriptions of successful methods it is assumed that a graph is connected. For double methods only one sequence of methods is given. An equivalent effect could be obtained by simply reversing the order of the method execution. For example, in Method 9(CT), executing Method 2(C) followed by Method 3(T) is equivalent to executing Method 3(T) followed by Method 2(C).

6.2.3.1 Analysis of Question 1

This question assumes that the learners understand that the optimum temperature is the best or most favourable temperature for photosynthesis, resulting in a maximum value for the rate of photosynthesis. To understand the significance of the instruction to ensure that neither light or carbon dioxide are in limited supplies implies an awareness that both of these variables can act as limiting factors. However, the specific instruction to maximise these variables attempts to ensure that learners who are unaware of this significance will still attempt to identify the optimum temperature with sensible C- and the L-values.

The application of a successful method will involve maximising the L-value and C-value and inspecting a graphical display to ascertain the optimum T-value. Method 1 can be applied with respect to both the C-sheet and the L-sheet:

- Method 1(C): select a C-sheet and maximise the C-value by locating the C-sheet at the maximum C-value position by executing an m_sheet(C) operation, maximise the L-value by locating the L-column at the maximum L-value position by executing an m_col(L) operation, and inspect the rate/T graph to determine the optimum temperature.
- Method 1(L): select an L-sheet and maximise the L-value by locating the L-sheet at the the maximum L-value position by executing an m_sheet(L) operation, maximise the C-value by locating the C-row at the maximum C-value position by executing an m_row(C) operation, and inspect the rate/T graph to find the optimum temperature.

Method 2 can be applied in a form which involves either an inspection of an animated rate/L graph or an animated rate/C graph:

- Method 2(T): select a T-sheet and maximise the C-value by locating the C-row at the maximum C-value position by executing an m_row(C) operation, inspect the resulting rate/L graph to determine the L-value corresponding to the maximum rate of photosynthesis, vary the T-value by executing m_sheet(T) operations, and observe the changes in the rate-value corresponding to the maximum L-value shown on the resulting instances of rate/L graphs to determine the optimum temperature.
- Method 2(T): select a T-sheet and maximise the L-value by locating the L-column at the maximum L-value position by executing an m_col(L) operation, inspect the resulting rate/C graph to determine the C-value corresponding to the maximum rate of photosynthesis, vary the T-value by executing m_sheet(T) operations, and observe the changes in the rate-value corresponding to the maximum C-value shown on the resulting instances of rate/C graphs to determine the optimum temperature.

Method 3 can be applied with respect to the L-sheet and the C-sheet:

- Method 3(C): select a C-sheet and maximise the C-value by locating the C-sheet at the maximum C-value position by executing an m_sheet(C) operation, select a T-row and inspect the resulting rate/L graph to identify the L-value corresponding to the maximum rate of photosynthesis, and execute m-row(T) operations to observe the changes in the rate-value corresponding to the maximum L-value shown on the resulting instances of rate/L graphs to determine the optimum temperature.

- Method 3(L): select an L-sheet and maximise the L-value by locating the L-sheet at the maximum L-value position by executing an m_sheet(L) operation, select a T-column and inspect the resulting rate/C graph to identify the C-value corresponding to the maximum rate of photosynthesis, and execute m_col(T) operations to observe the changes in the rate-value corresponding to the maximum C-value shown on the resulting instances of rate/L graphs to determine the optimum temperature.

The action strings for each of these successful methods are shown in Table 6.2.

Table 6.2: Action strings for successful methods for Question 1

Method	Action string
1(C)	ll s_sheet(C) l m_sheet(C) ll s_col(L) l m_col(L) ll inspect_graph(r/T)
1(L)	ll s_sheet(L) l m_sheet(L) ll s_row(C) l m_row(C) ll inspect_graph(r/T)
2(T)	ll s_sheet(T) ll s_row(C) l m_row(C) ll inspect_graph(r/L) ll repeat [m_sheet(T) l inspect_graph(r/L)]
2(T)	ll s_sheet(T) ll s_col(L) l m_col(L) ll inspect_graph(r/C) ll repeat [m_sheet(T) l inspect_graph(r/C)]
3(C)	ll s_sheet(C) l m_sheet(C) ll s_row(T) ll inspect_graph(r/L) ll repeat [m_row(T) l inspect-graph(r/L)]
3(L)	ll s_sheet(L) l m_sheet(L) ll s_col(T) ll inspect_graph(r/C) ll repeat [m_col(T) l inspect-graph(r/C)]

6.2.3.2 Analysis of Question 2

Question 2 is concerned with an exploration of the effects on the rate of photosynthesis of changing the C-value or the L-value. The question states that these explorations should be conducted at the optimum temperature. No instructions are given to maximise the L-value while the effect of varying the C-value is being explored, or to maximise the C-value while the effect of varying the L-value is being explored. This omission can be interpreted as an expectation that the learners will conduct their explorations with the the L-value or C-value set at an arbitrary convenient value. In this case the influence of the "third" variable would not be explicitly considered, indicating that it was thought that the learners will not consider the possible interaction of this variable with the other variables, and more specifically that learners will not appreciate the significance of setting the C-value or the L-value at a non limiting value.

Methods 1, 2 and 3 can be used in task-truncated forms to answer this question. In Method 1 the m_sheet(C) or m_sheet(L) operation does not need to be executed to maximise respectively the C- or L-value. However, it may be necessary to execute one of these operations to locate the C- or L-sheet in a convenient position. For example, if the position of the sheet defaults to the minimum value location a row or column consisting of a set of zero rate values may result, leading to the display of an instance of a "null" rate/variable graph. The task-truncated versions of Method 1 for exploring the effect of changing the light intensity are the following:

- Method 1_{tt}(C): select a C-sheet and if necessary move it to a convenient position, set the temperature at the optimum value by locating the T-row at the position corresponding to the optimum T-value by executing an m_row(T) operation, and inspect the resulting rate/L graph to observe the variation of the rate with the L-value.
- Method 1_{tt}(T): select a T-sheet and locate it at the optimum temperature position, select a C-row, execute an m_row(C) operation to choose a convenient C-value, and inspect the rate/L graph to observe the variation of the rate with the L-value.

The task-truncated versions for exploring the effect of changing the carbon dioxide concentration are the following:

- Method 1_{tt}(L): select an L-sheet and if necessary move it to a convenient position, set the temperature at the optimum value by locating the T-column at the position corresponding to the optimum T-value by executing an m_col(T) operation, and inspect the resulting rate/C graph to observe the variation of the rate with the C-value.
- Method 1_{tt}(T): select a T-sheet and locate it at the optimum temperature position, select an L-column, execute an m_col(L) operation to choose a convenient L-value, and inspect the rate/C graph to observe the variation of the rate with the C-value.

Method 2 would be truncated because it is not necessary to inspect the rate/variable graph after a row or column has been selected as attention is not being focused on the maximum C-value or L-value. The task-truncated versions of this method become:

- Method 2_{tt}(L): select an L-sheet, locate the T-column in the position corresponding to the optimum temperature, vary the L-value by executing m_sheet(L) operations, and observe the changes in the rate/C graph.

- Method $2_{tt}(C)$: select a C-sheet, locate the T-row in the position corresponding to the optimum temperature, vary the C-value by executing m_sheet(C) operations, and observe the changes in the rate/L graph.

Method 3 would be task-truncated due to the rate/variable graph not being inspected after an s_row(C) or s_col(L) operation is executed, leading to the following truncated versions:

- Method $3_{tt}(T)$: select a T-sheet and locate it at the position corresponding to the optimum temperature, select a C-row, execute m_row(C) operations, and inspect the resulting instances of rate/L graphs to observe the variation of the rate with the C-value.
- Method $3_{tt}(T)$: select a T-sheet and locate it at the position corresponding to the optimum temperature, select an L-column, execute m_col(L) operations, and inspect the resulting instances of rate/C graphs to observe the variation of the rate with the L-value.

The action strings for these truncated successful methods are shown in Table 6.3.

Table 6.3: Action strings for full and truncated successful methods for Question 2

Method	Action string
$1_{tt}(C)$	ll s_sheet(C) l (m_sheet(C)) ll s_col(T) l m_col(T) ll inspect_graph(r/L)
$1_{tt}(L)$	lls_sheet(L) l (m_sheet(L)) ll s_row(T) l m_row(T) ll inspect_graph(r/C)
$1_{tt}(T)$	ll s_sheet(T) l m_sheet(T) ll s_col(L) l (m_col(L)) ll inspect_graph(r/C)
$1_{tt}(T)$	ll s_sheet(T) l m_sheet(T) ll s_row(C) l (m_row(C)) ll inspect_graph(r/L)
$2_{tt}(C)$	ll s_sheet(C) ll s_row(T) l m_row(T) ll repeat [m_sheet(C) l inspect_graph(r/L)]
$2_{tt}(L)$	ll s_sheet(L) ll s_col(T) l m_col(T) ll repeat [m_sheet(L) l inspect_graph(r/C)]
$3_{tt}(T)$	ll s_sheet(T) l m_sheet(T) ll s_row(C) ll repeat [m_row(C) l inspect-graph(r/L)]
$3_{tt}(T)$	ll s_sheet(T) l m_sheet(T)ll s_col(L)ll repeat [m_col(L) l inspect-graph(r/C)]

6.2.3.3 Analysis of Question 3

In this question the learners are asked to find the optimum value of the carbon dioxide concentration or the light intensity. There are no explicit instructions to ensure that neither light nor carbon dioxide are in limited supplies or to set the temperature at its optimum value. However, the experience gained in answering Questions 1 and 2 should indicate to students that the T-value should be set to the optimum value, and that the C-value should be maximised when the optimum L-value is being determined and vica versa. If these assumptions are made Methods 1, 2 and 3 will be applied in full.

Applications of Method 1 to determine the optimum C-value would be as follows:

- Method 1(L): select a L-sheet and maximise the L-value by locating the L-sheet at the position corresponding to the maximum L-value by executing an m_sheet(L) operation, optimise the T-value by locating the T-column at the position corresponding to the optimum T-value by executing an m_col(T) operation, and inspect the resulting rate/C graph to determine the optimum C-value.
- Method 1(T): select a T-sheet and locate it at the optimum temperature position, select an L-column, execute an m_col(L) operation to maximise the L-value, and inspect the rate/C graph to determine the optimum C-value.

Applications of Method 1 to determine the optimum L-value would be as follows:

- Method 1(C): select a C-sheet and maximise the C-value by locating the C-sheet at the position corresponding to the maximum C-value by executing an m_sheet(C) operation, optimise the T-value by locating the T-row at the position corresponding to the optimum T-value by executing an m_row(T) operation, and inspect the resulting rate/L graph to determine the optimum L-value.
- Method 1(T): select a T-sheet and locate it at the optimum temperature position, select an C-row, execute an m_row(C) operation to maximise the C-value, and inspect the rate/L graph to determine the optimum L-value.

Method 2 would take the following forms:

- Method 2(C): select a C-sheet and optimise the T-value by locating the T-row at the optimum T-value position by executing an m_row(T) operation, inspect the resulting rate/L graph to determine the L-value corresponding to the maximum rate of photosynthesis, vary the C-value by executing m_sheet(C) operations, and observe the changes in the rate-value corresponding to the maximum L-value shown on the resulting instances of rate/L graphs to determine the optimum C-value.
- Method 2(L): select a L-sheet and optimise the T-value by locating the T-column at the optimum T-value position by executing an m_col(T) operation, inspect the resulting rate/C graph to determine the C-value corresponding to the maximum rate of photosynthesis, vary the L-value by executing m_sheet(L) operations, and observe the changes in the rate-value corresponding to the maximum C-value shown on the resulting instances of rate/C graphs to determine the optimum L-value.

Method 3 can be applied to the T-sheet to determine the optimum C-value and the L-value:

- Method 3(T): select a T-sheet and optimise the T-value by locating the T-sheet at the optimum temperature position, select a C-row and inspect the resulting rate/L graph to identify the L-value corresponding to the maximum rate, and execute m_row(C) operations to observe the changes in the rate corresponding to the maximum L-value shown on the rate/L graphs to determine the optimum C-value.
- Method 3(T): select a T-sheet and optimise the T-value by locating the T-sheet at the optimum temperature position, select an L-column and inspect the resulting rate/C graph to identify the C-value corresponding to the maximum rate, and execute m_col(L) operations to observe the changes in the rate corresponding to the maximum C-value shown on the instances of rate/L graphs to determine the optimum L-value.

The action strings for each of these successful methods are shown in Table 6.4.

Table 6.4: Action strings for successful methods for Question 3

Method	Action string
1(C)	ll s_sheet(C) l m_sheet(C) ll s_row(T) l m_row(T) ll inspect_graph(r/L)
1(L)	ll s_sheet(L) l m_sheet(L) ll s_col(T) l m_col(T) ll inspect_graph(r/C)
1(T)	ll s_sheet(T) l m_sheet(T) ll s_col(L) l m_col(L) ll inspect_graph(r/C)
1(T)	ll s_sheet(T) l m_sheet(T) ll s_row(C) l m_row(C) ll inspect_graph(r/L)
2(C)	ll s_sheet(C) ll s_row(T) l m_row(T) ll inspect_graph(r/L) ll repeat [m_sheet(C) l inspect_graph(r/L)]
2(L)	ll s_sheet(L) ll s_col(T) l m_col(T) ll inspect_graph(r/C) ll repeat [m_sheet(L) l inspect_graph(r/C)]
3(T)	ll s_sheet(T) l m_sheet(T) ll s_row(C) l inspect_graph(r/L) ll repeat [m_row(C) l inspect-graph(r/L)]
3(T)	ll s_sheet(T) l m_sheet(T) ll s_col(L) l inspect_graph(r/C) ll repeat [m_col(L) l inspect-graph(r/C)]

6.2.3.4 Analysis of Question 4

Question 4 is concerned with the difference in the effect of increasing levels of light or carbon dioxide on the rate of photosynthesis. The specific instruction in Question 2 to explore the effects of changing the C-value or the L-value at the optimum temperature indicates that this comparison is best made at this temperature. As with Question 2 there are no instructions which imply that the limiting effects of either the C-value or the L-value should be considered while the effect of varying the other variable is being considered.

The comparison can be made by comparing a rate/L graph and a rate/C graph at the optimum temperature. If it assumed that the L-value and C-value are not maximised Methods 4_{tt}(LC), 4_{tt}(LT), 4_{tt}(CT) and 4_{tt}(TT) can be applied to display these graphs at the same time in separate windows:

- Method 4_{tt}(LC): apply Method 1_{tt}(L), if necessary connect a second graph window, apply Method 1_{tt}(C), and compare the resulting rate/C and rate/L graphs.
- Method 4_{tt}(LT): apply Method 1_{tt}(L), if necessary connect a second graph window, apply Method 1_{tt}(T) with a row chosen, and compare the rate/C and rate/L graphs.
- Method 4_{tt}(CT): apply Method 1_{tt}(C), if necessary connect a second graph window, apply Method 1_{tt}(T) with a column chosen, and compare the rate/L and rate/C graphs.
- Method 4_{tt}(TT) apply Method 1_{tt}(T) with a column selected, if necessary connect a second graph window, apply Method 1_{tt}(T) with a row selected, and compare the resulting rate/C and rate/L graphs.

Method 5 can also be applied to compare the effects of changing the L-value and the C-value. When this method is applied animated rate/L and rate/C graphs a viewed in turn. As with the application of Method 4, the graphs should correspond to the optimum temperature, and the method would be truncated as maximum C- and L-values would not be set. The task-truncated applications of Method 5 would take the form:

- Method 5_{tt}(LC): apply Method 2_{tt}(L) and observe the sequence of rate/C graphs as the L-value is changed by executing m_sheet(L) operations, apply Method 2_{tt}(C) and observe the sequence of rate/L graphs as the C-value is changed by executing m_sheet(C) operations, and compare the changes in rate/C and the rate/L graphs.

Method 6 can also be applied to compare the effects of changing the L-value and the C-value. When this method is applied animated rate/L and rate/C graphs a viewed in turn. The graphs should correspond to the optimum temperature, and the method would be truncated as maximum C- and L-values would not be set. The truncated application of Method 6 would take the form:

- Method 6_{tt}(TT): apply Method 3_{tt}(T) and observe the sequence of rate/C graphs as the L-value is changed by executing m_col(L) operations, apply Method 3_{tt}(T) and observe the sequence of rate/L graphs as the C-value is changed by executing m_row(C) operations, and compare the changes in the rate/C and the rate/L graphs.

Method 7 can also be applied to compare the effects of changing the L-value and the C-value. When this method is applied an animated rate/L graph is observed to investigate the effect of varying the C-value and a static rate/L graph is inspected to explore the effect of changing the L-value, or an animated rate/C graph is observed to investigate the effect of varying the L-value and a static rate/C graph is inspected to explore the effect of changing the C-value. The graphs should correspond to the optimum temperature, that is the row or column corresponding to the optimum temperature should be chosen in the applications of the component Methods 1 and 2 as appropriate. The method would be truncated as maximum C- and L-values would not be considered. The truncated applications of Method 7 would take the form:

- Method 7_{tt}(CC): apply Method 1_{tt}(C) and observe a static rate/L graph, apply Method 2(C)$_{tt}$ and observe the sequence of rate/L graphs as the C-value is changed by executing m_sheet(C) operations, and compare the changes in the animated and static rate/L graphs.
- Method 7_{tt}(LL): apply Method 1_{tt}(L) and observe a static rate/C graph, apply Method 2(L)$_{tt}$ and observe the sequence of rate/C graphs as the L-value is changed by executing m_sheet(L) operations, and compare the changes in the animated and static rate/C graphs.
- Method 7_{tt}(TL): apply Method 1_{tt}(T) with a column selected and observe a static rate/C graph, apply Method 2(L)$_{tt}$ and observe the sequence of rate/C graphs as the L-value is changed by executing m_sheet(L) operations, and compare the changes in the animated and static rate/C graphs.
- Method 7_{tt}(TC): apply Method 1_{tt}(T) with a row selected and observe a static rate/L graph, apply Method 2(L)$_{tt}$ and observe the sequence of rate/L graphs as the C-value is changed by executing m_sheet(C) operations, and compare the changes in the animated and static rate/L graphs.

Method 8 can also be applied to compare the effects of changing the L-value and the C-value. As in Method 7, when this method is applied an animated rate/L graph is observed to investigate the effect of varying the C-value and a static rate/L graph is inspected to explore the effect of changing the L-value, or an animated rate/C graph is observed to investigate the effect of varying the L-value and a static rate/C graph is inspected to explore the effect of changing the C-value. The graphs should correspond to the optimum temperature, that is the row or column corresponding to the optimum temperature should be chosen in the application of a component Method 1 as appropriate, and the T-sheet should be located in the optimum temperature position for for the application of the component Method 3_{tt}(T). The method would be truncated as maximum C- and L-values would not be considered. The truncated applications of Method 8 would take the form:

- Method 8_{tt}(CT): apply Method 1_{tt}(C) and observe a static rate/L graph, apply Method 3_{tt}(T) and observe the sequence of rate/L graphs as the C-value is changed by executing m_row(C) operations, and compare the changes in the animated and static rate/L graphs.
- Method 8_{tt}(LT): apply Method 1_{tt}(L) and observe a static rate/C graph, apply Method 3_{tt}(T) and observe the sequence of rate/C graphs as the L-value is changed by executing m_col(L) operations, and compare the changes in the animated and static rate/L graphs.
- Method 8_{tt}(TT): apply Method 1_{tt}(T) with a column selected and observe a static rate/C graph, apply Method 3_{tt}(T) and observe the sequence of rate/C graphs as the L-value is changed, and compare the changes in the animated and static graphs.
- Method 8_{tt}(TT): apply Method 1_{tt}(T) with a row selected and observe a static rate/L graph, apply Method 3_{tt}(T) and observe the sequence of rate/L graphs as the C-value is changed, and compare the changes in the animated and static graphs.

Method 9 can also be applied to compare the effects of changing the L-value and the C-value. Animated rate/L and rate/C graphs a viewed in turn. The graphs should correspond to the optimum temperature, that is the row or column corresponding to the optimum temperature should be selected for the application of a component Method 2(C) or 2(L) respectively, and the T-sheet should be located in the optimum temperature position for the component Method 3(T). The method would be truncated as maximum C- and L-values would not be set. The truncated applications of Method 9 would take the form:

- Method 9_{tt}(CT): apply Method 2_{tt}(C) and observe the sequence of rate/L graphs as the C-value is changed by executing m_sheet(C) operations, apply Method 3_{tt}(T) and observe the sequence of rate/C graphs as the L-value is changed by executing m_col(L) operations, and compare the changes in the rate/C and the rate/L graphs.
- Method 9_{tt}(LT): apply Method $2(L)_{tt}$ and observe the sequence of rate/C graphs as the L-value is changed by executing m_sheet(L) operations, apply Method 3_{tt}(T) and observe the sequence of rate/C graphs as the C-value is changed by executing m_row(C) operations, and compare the changes in the rate/C and the rate/L graphs.

The action sub-strings for each of these successful methods are shown in Table 6.5.

Table 6.5: Action strings for successful methods for Question 4

<table>
<tr><th>Method</th><th>Action string</th></tr>
<tr><td>4_{tt}(LC)</td><td>|| s_sheet(L) | (m_sheet(L)) || s_col(T) | m_col(T) || inspect_graph(r/C)
|| con_graph || s_sheet(C) | (m_sheet(C)) || s_row(T) | m_row(T)
|| inspect_graph(r/L)</td></tr>
<tr><td>4_{tt}(LT)</td><td>|| s_sheet(L) | (m_sheet(L)) || s_col(T) | m_col(T) || inspect_graph(r/C)
|| con_graph || s_sheet(T) | m_sheet(T) || s_row(C) | (m_row(C))
|| inspect_graph(r/L)</td></tr>
<tr><td>4_{tt}(CT)</td><td>|| s_sheet(C) | (m_sheet(C)) || s_row(T) | m_row(T) || inspect_graph(r/L)
|| con_graph || s_sheet(T) | m_sheet(T) || s_col(L) | (m_col(L))
|| inspect_graph(r/C)</td></tr>
<tr><td>4_{tt}(TT)</td><td>|| s_sheet(T) | m_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C)
|| con_graph || s_row(C) | (m_row(C)) || inspect_graph(r/L)</td></tr>
<tr><td>5_{tt}(LC)</td><td>|| s_sheet(L) || s_col(T) | m_col(T) || repeat [m_sheet(L) | inspect_graph(r/C)]
|| s_sheet(C) || s_row(T) | m_row(T) || repeat [m_sheet(C) | inspect_graph(r/L)]</td></tr>
<tr><td>6_{tt}(TT)</td><td>|| s_sheet(T) | m_sheet(T) || s_row(C) || repeat [m_row(C) | inspect-graph(r/L)]
|| s_col(L) || repeat [m_col(L) | inspect-graph(r/C)]</td></tr>
<tr><td>7_{tt}(CC)</td><td>|| s_sheet(C) (| m_sheet(C)) || s_row(T) | m_row(T) || inspect_graph(r/L)
|| repeat [m_sheet(C) | inspect_graph(r/L)]</td></tr>
<tr><td>7_{tt}(LL)</td><td>|| s_sheet(L) | (m_sheet(L)) || s_col(T) | m_col(T) || inspect_graph(r/C)
|| repeat [m_sheet(L) | inspect_graph(r/C)]</td></tr>
<tr><td>7_{tt}(TL)</td><td>|| s_sheet(T) | m_sheet(T) || s_col(L) | (m_col(L)) || inspect_graph(r/C)
|| s_sheet(L) || s_col(T) | m_col(T) || repeat [m_sheet(L) | inspect_graph(r/C)]</td></tr>
<tr><td>7_{tt}(TC)</td><td>|| s_sheet(T) | m_sheet(T) || s_row(C) | (m_row(C)) || inspect_graph(r/L)
|| s_sheet(C) || s_row(T) | m_row(T) || repeat [m_sheet(C) | inspect_graph(r/C)]</td></tr>
<tr><td>8_{tt}(CT)</td><td>|| s_sheet(C) (| m_sheet(C)) || s_row(T) | m_row(T) || inspect_graph(r/L)
|| s_sheet(T) | m_sheet(T) || s_row(C) || repeat [m_row(C) | inspect-graph(r/L)]</td></tr>
<tr><td>8_{tt}(LT)</td><td>|| s_sheet(L) | (m_sheet(L)) || s_col(T) | m_col(T) || inspect_graph(r/C)
|| s_sheet(T) | m_sheet(T) || s_col(L) || repeat [m_col(L) | inspect-graph(r/C)]</td></tr>
<tr><td>8_{tt}(TT)</td><td>|| s_sheet(T) | m_sheet(T) || s_row(C) | (m_row(C)) || inspect_graph(r/L)
|| repeat [m_row(C) | inspect-graph(r/L)]</td></tr>
<tr><td>8_{tt}(TT)</td><td>|| s_sheet(T) | m_sheet(T) || s_col(L) | (m_col(L)) || inspect_graph(r/C)
|| repeat [m_col(L) | inspect-graph(r/C)]</td></tr>
<tr><td>9_{tt}(CT)</td><td>||s_sheet(C) ||s_row(T) | m_row(T) ||repeat [m_sheet(C) | inspect_graph(r/L)]
||s_sheet(T) | m_sheet(T) ||s_col(L) ||repeat [m_col(L) | inspect-graph(r/C)]</td></tr>
</table>

Table 6.5: Action strings for successful methods for Question 4 (continued)

9_{tt}(LT)	||s_sheet(L) ||s_col(T) | m_col(T) ||repeat [m_sheet(L) | inspect_graph(r/C)]
	||s_sheet(T) | m_sheet(T) ||s_row(C) ||repeat [m_row(C) | inspect-graph(r/L)]

6.2.3.5 Analysis of Question 5

Question 5 refers to the maximum rate of photosynthesis shown in the data. In this state the temperature will be at the optimum level, and the the levels of carbon dioxide and light intensity will be at a maximum value. At this maximum rate the question asks which factor should be increased more in an attempt to increase the rate further. As the temperature is at the optimum value, any change in the T-value will not increase the rate further. While the L-value is close to a non-limiting value, the C-value is still at a limiting value. Therefore, the level of carbon dioxide should be increased. This can be illustrated by applying Methods 4(LC), 4(LT), 4(CT) or 4(TT):

- Method 4(LC): apply Method 1(L), if necessary connect a second graph window, apply Method 1(C), and compare the resulting rate/C and rate/L graphs.
- Method 4(LT): apply Method 1(L), if necessary connect a second graph window, apply Method 1(T) with a row selected, and compare the resulting graphs.
- Method 4(CT): apply Method 1(C), if necessary connect a second graph window, apply Method 1(T) with a row selected, and compare the resulting graphs.
- Method 4(TT): apply Method 1(T) with a row selected, if necessary connect a second graph window, apply Method 1(T) with a column selected, and compare the resulting graphs.

The action sub-strings for these successful methods are shown in Table 6.6.

6.3 Summary

It is clear from the analysis of the Users' Guide that an ability to use a sub-set of *Windows* is essential. However, the analysis of questionnaire responses from Masters students indicated that handling *Windows* may be problematic, particularly for novice users. However, there is an indication that such problems can be overcome relatively easily.

Analysis of the questionnaire responses indicates that the datacube representation could cause some conceptual problems; an indication supported by a comment by Thompson:

> The level of understanding about the purpose of this cube went no deeper than it being a navigational aid [...]. It may have been that the nature of the task [..] was too closed [...]. If they had been given a much more open ended assignment, might they have developed a deeper understanding of the use of the cube? (p. 50)

Table 6.6: Action strings for successful methods for Question 5

Method	Action string
4(LC)	ll s_sheet(L) l m_sheet(L) ll s_col(T) l m_col(T) ll inspect_graph(r/C) ll con_graph ll s_sheet(C) l m_sheet(C) ll s_row(T) l m_row(T) ll inspect_graph(r/L)
4(TC)	ll s_sheet(T) l m_sheet(T) ll s_col(L) l m_col(L) ll inspect_graph(r/C) ll con_graph ll s_sheet(C) l m_sheet(C) ll s_row(T) l m_row(T) ll inspect_graph(r/L)
4(TL)	ll s_sheet(T) l m_sheet(L) ll s_row(C) l m_row(C) ll inspect_graph(r/L) ll con_graph ll s_sheet(L) l m_sheet(L) ll s_col(T) l m_col(T) ll inspect_graph(r/C)
4(TT)	ll s_sheet(T) l m_sheet(T) ll s_row(C) l m_row(C) ll inspect_graph(r/L) ll con_graph ll s_col(L) l m_col(L) ll inspect_graph(r/C)

The problems identified in manipulating the graphing features by the group of Masters students were also confirmed by Thompson's report.

Some indication of the problems of observing students' use of *Bioview* was documented by Thompson and it seems that a video based technique is necessary to capture the essential features of the human-computer interaction.

From a review of the content of the textbooks that the teacher-students had access to it is clear that they were expected to understand the concept of a limiting factors and interpret rate variable graphs in terms of this concept. They were also expected to know about the concept of the optimum temperature. These ideas are reflected in the worksheet questions; Question 1 deals specifically with the concept of optimum temperature, Question 2 is concerned with the effects of changing light intensity and carbon dioxide levels, and Questions 3, 4 and 5 are concerned with limiting values.

In addition to identifying the task related aspects inferred in the worksheet questions it is also possible to identify successful methods for the use of *Bioview* to answer these questions. Three basic successful methods have been identified which can all be used to support attempts to answer Questions 1-3. These methods can be used in combination to provide methods applicable when Questions 4 and 5 are attempted.

Chapter 7

Analysis of laboratory observations

In this chapter the video records of the three sessions in which the use of *Bioview* was observed are analysed from an interaction perspective and a task perspective. Each session has been divided into a series of task related episodes. Typically each episode corresponds to an attempt to answer one of the set questions. However, each session included an initial session which was concerned with a demonstration of the general features of the program without reference to its use to answer a specific question. In addition the final episode in the Ruth/Tom session was concerned with a general discussion of the merits of the software. The applicability of an extended version of the GOMS model in providing an appropriate framework for analysing the human-computer interaction in each of the three sessions is also considered.

7.1 Interaction with *Bioview* from a system perspective

Interaction histories for each of the three sessions have been prepared (see Appendix 7). These histories have been used to identify the method selection sequence for each session and, in conjunction with the episode record sheets, to identify occasions when there was a change of active system sub-register. In addition the action strings corresponding to each application of a successful method (see Appendix 8) have been used to determine the action string lengths for each episode. The results of these analyses are shown in Table 7.1.

Episodes for which there was a net change of active system sub-register have been identified and indicated by "Δ". It was also possible for changes of active system sub-register to take place with no net effect within an episode. This may have occurred due to the effects of changes in unit-tasks in an episode cancelling out, or through the effects of changes within a unit-task cancelling. These episodes have been identified and indicated by "∂". The action strings shown in the task-episode tables shown in Appendix 8 have been used to determine observed action string lengths (n_o), string lengths which are corrected for manipulation errors (n_c), and string lengths which are reduced to account for the execution of window manipulation operations (notably scope_win operations executed to enable the whole range of a datasheet to be used) and graph selection operations (n_r). The error free reduced action string lengths (n_e) for successful methods have been determined by inspecting the action string look-up tables shown in Appendix 3. The quoted value of n_e for each episode has been determined by matching the observed implementation of the

Table 7.1: Method action string lengths for successfully applied methods

Episode	Meth.	Trun.	Sub-reg. change	Number of operations (action string lengths)					
				n_o	n_c	n_r	n_e	n_o-n_c	n_r-n_e
D2	1(C)		∂	14	14	9	5	0	4
D3A	3(T)	t	Δ	30	25	5	5	5	0
D3B	3(T)	t d	∂	12	12	6	4	0	2
D4A	1(T)	d	o	2	2	1	1	0	0
D4B	3(T)	t d	o	5	5	2	2	0	0
D5	6(TT)	t d	o	17	17	5	5	0	0
D6	Recall		o	-	-	-	-	-	-
D7	3(L)	t d	Δ	6	6	4	3	0	1
D8	2(T)	t	Δ	13	13	9	5	0	4
D9A	3(T)	t d	Δ	24	20	5	3	4	2
D9B	3(T)	t d	o	10	10	4	3	0	1
D10	3(T)	t d	o	14	14	3	3	0	0
D11	6(TT)	t d	o	38	36	9	8	2	1
D12	Recall	-	o	-	-	-	-	-	-
S2	1(L)	d	∂	10	8	5	4	2	1
S3A	3(T)	t d	Δ	9	9	4	4	0	0
S3B	3(T)	t d	o	9	9	3	3	0	0
S4	3(T)	t d	o	7	7	2	2	0	0
S5	6(TT)	t d	o	3	3	3	3	0	0
S6	8(LL)	t d	o	1	1	1	1	0	0
S7	2(T)	d	∂	10	8	8	4	2	4
S8A	3(T)	t d	o	9	9	2	2	0	0
S8B	3(T)	t d	o	23	23	3	3	0	0
S9A	1(T)	d	o	1	1	1	1	0	0
S9B	3(T)	t d	o	15	12	2	2	3	0
S10	Recall		o	-	-	-	-	-	-
S11	2(C)	t	Δ	8	8	7	5	0	2
R2	1(L)	d	o	3	3	2	2	0	0
R3A	3(T)	t d	Δ	10	5	4	4	5	0
R3B	3(L)	t d	Δ	3	3	3	3	0	0
R4	1(T)	d	Δ	12	11	6	3	1	3
R5	Recall	-	o	-	-	-	-	-	-
R6A	{4}	-	Δ	10	9	3	-	1	-
R6B	{4}	-	Δ	9	5	1		4	-
R6C	{4}	-	∂	18	17	17	-	1	-
R6D	7(TT)	-	o	21	21	3	3	0	0
R6E	{Idio}	-	Δ	51	46	46	-	5	-
R6F	{Idio}	-	Δ	19	16	16	-	0	-
R6G	{Idio}	-	∂	16	16	16	-	0	-

method and the error free appropriately truncated version of the method. By definition, values for n_e and n_r-n_e are not given for idiosyncratic methods. The difference between n_o

and n_c gives the number of errors in each episode and the difference between n_r and n_e indicates the difference in the observed performance and expert performance.

In the following sections aspects of human-computer interaction which occur in the three sessions are discussed in terms of (i) method truncation, (ii) the origin and significance of errors, (iii) differences in observed and "expert" execution of successful methods, and (iv) the application of unsuccessful methods.

7.1.1 Method truncation

As discussed in Sections 4.3.2 and 5.2.1.2 method truncation occurs when the number of operations required to successfully implement a method is reduced. Truncation can be task related or display related. It is clear from Table 7.1 that the majority of the episodes which featured the use of successful methods involved method truncation. Task related truncation is discussed in Section 7.2. Display related truncation is discussed in detail in this section.

Table 7.2 gives details of the system related truncation observed in the episodes that featured the application of a successful method. The value of $\Delta n_e(f)$ gives the number of unnecessary operations associated with full display-truncation and $\Delta n_e(d)$ gives the observed number of non-executions of operators. A value of zero for $\Delta n_e(d)$ indicates that no display related truncation took place, and a value of zero for the difference between $\Delta n_e(f)$ and $\Delta n_e(d)$ indicates that full display related truncation took place.

Although Episode D2 did not feature a truncation of Method 1, partial truncation of this method was evident in Episodes S2, R2 and R4, and full truncation was observed in Episodes D4A and S9A. All five truncated applications of Method 1 featured the omission of an s_row/col operation, three of these episodes involved the omission of an s_sheet operation, and in four episodes it was unnecessary to move the position of the active datasheet. In Episode R2 Ruth clearly appreciated that Method 1(L) could be applied in a truncated form. This episode started with the L-sheet located in the position corresponding to the maximum L-value, making Unit-task 1 redundant. She informed Tom that the light intensity was already set at a maximum and that the next step should be to move the currently selected C-row to the position corresponding to the maximum level of carbon dioxide:

> R: So at the moment got maximum light intensity, which is at 50 and carbon dioxide. We need to, if you go on [*scope_win operation executed*] ... and pull it down, [*referring to a C-row*] should go down, if keep on going down. Go to 0.1 [*referring to the maximum value of carbon dioxide*] and click on it.

Delia and Sharon simply advised the inspection of currently displayed graphs at the start of Episodes D4A and S9A respectively, as illustrated in Episode D4A:

D: Right. Basically if you do it type, [*referring to the type menu item in the graph_window (2) menu bar*] do bar graph for that one. [*s_graph(b) operation executed*] Like rate, where rate of photosynthesis is the greatest, so if look at the graph the light intensity is increasing along the bottom, and where is rate of photosynthesis the greatest?

Table 7.2: Display related truncation observed in the application of successful methods

Episode	Method	Display truncation				$\Delta n_e(f)$	$\Delta n_e(d)$	$\Delta n_e(f)-\Delta n_e(d)$
		s_sheet	m_sheet	s_r/c	m_r/c			
D2	1(C)					4	0	4
D4A	1(T)	•	•	•	•	4	4	0
S2	1(L)			•		4	1	3
S9A	1(T)	•	•	•	•	4	4	0
R2	1(L)	•	•	•		4	3	1
R4	1(T)		•	•		4	2	2
D8	2(T)					3	0	3
S7	2(T)			•		3	1	2
S11	2(C)					3	0	3
D3A	3(T)					3	0	3
D3B	3(T)		•			3	1	2
D4B	3(T)	•	•	•		3	3	0
D7	3(L)		•	•		3	2	1
D9A	3(T)		•	•		3	2	1
D9B	3(T)	•	•			3	2	1
D10	3(T)	•	•			3	2	1
S3A	3(T)			•		3	1	2
S3B	3(T)	•	•			3	2	1
S4	3(T)	•	•	•		3	3	0
S8A	3(T)	•	•	•		3	3	0
S8B	3(T)	•	•			3	2	1
S9B	3(T)	•	•	•		3	3	0
R3A	3(T)			•		3	1	2
R3B	3(L)		•	•		3	2	1
D5	6(TT)	••	••	•		5	5	0
D11	6(TT)	••	••	•		5	5	0
S5	6(TT)	••	••	•		5	5	0
R6D	7(TT)	••	•	••	••	7	7	0
S6	8(LL)	••	••	•	••	7	7	0

This graph corresponded to the optimum temperature, as the datasheet linked to this temperature was chosen in the previous episode, and to the maximum level of carbon dioxide, as the C-row corresponding to this value was highlighted. It seems that the state of the display prompted Delia to advise Alice to apply a fully display related truncated version of Method 1(T).

Method 2 was only applied in three episodes. Episodes D8 and S11 involved full applications of Method 2, and the partially truncated application of this method in Episode S7 only featured the omission of an s_row operation. Although Episode D8 was very confused, with the inclusion of an irrelevant unit-task concerned with the observation of the effect of varying the light intensity, the unit-tasks concerned with maximising the level of carbon dioxide and varying the temperature were clear full implementations.

Method 3 was applied in 15 episodes, making it the most frequently selected method. The application of the method was partially truncated in 10 episodes, and fully truncated in four episodes. Twelve episodes featured a truncation of Unit-task 1. In eight of these episodes this unit-task was completely unnecessary, and in the remaining four episodes the m_sheet operation was omitted. Episode D9A illustrated that Delia appreciated that the T-sheet displayed at the start of the episode would truncate an application of Method 3(T). At the beginning of this episode Delia verbally confirmed that the temperature needed to be set at the optimum value. Although the T-sheet was already located in this position Alice selected the C-sheet and executed an m_sheet(C) operation. Delia helped Alice to reset the datasheet location to the optimum temperature position :

> D: OK, number two. At this optimum temperature look at the way in which varying the light and carbon dioxide levels affects the rate of photosynthesis. So for this one you've got to have optimum temperature.
>
> A: Which is 30 degrees.
>
> D: Yeah
>
> A: So do I just leave it like this? [*referring to the graph window(1) display*]
>
> D: Yes, leave it at 30 degrees, but got to vary light and carbon dioxide.
>
> A: So got to change it here? [*s_sheet(C) operation executed by clicking on the C-value scroll box and executing an m_sheet(C) operation*]
>
> D: So, if you go back to that, [*s_sheet(T) operation executed*] this is set at 30, yeah.

Nine episodes featured the omission of an s_row/col operation when Method 3 was applied. The influence of this type of display related truncation was illustrated in Episode S9B. This episode featured a fully truncated application of Method 3(T). Uri was asked by Sharon which variable (light intensity or level of carbon dioxide) he wanted to observe the effects of changing. He replied "not bothered". Sharon suggested carbon dioxide, but in fact she gave instructions that resulted in the exploration of the effect of changing the light intensity. It seems likely that the current selection of a column on the T-sheet led Sharon to instinctively truncate the application of Method 3.

In all three episodes that featured an application of Method 6(TT) the T-sheet was already selected and located at the start of the episode in the position corresponding to the optimum temperature. This led to fully truncated applications of this method. Likewise the active state at the start of Episode R6D and S6 of respectively a T-sheet and an L-sheet was instrumental in the application of a fully truncated version of Method 7(TT) and Method 8(LL).

The extensive use of display related truncation indicates that the current screen configuration, and by implication the state of the system, had a significant effect on method selection. This was illustrated particularly by the applications of Method 3(T). The vast majority of these applications were truncated by the omission of an s_sheet(T) operation, indicating that the display of a T-sheet at the start of an episode strongly encouraged the choice of a method based on the use of an active T-sheet. Similarly the fully truncated form of all of the applications of Method 6(TT) and the single applications of Methods 7(TT) and 8(LL) indicate the significance of the initial display state on method selection.

The use of different methods to answer the same question in different episodes also indicates that the state of the system at the start of an episode was highly influential in method selection. For example, different methods were selected in Episode S2 and S7 to answer Question 1; as appreciated by Sharon when she described the method adopted in Episode S7 as a "different way of finding it than we did before". Method 1(L) was applied in Episode S2 and Method 2(T) was adopted in Episode S7. The T-sheet was selected in the position corresponding to the optimum temperature at the start of Episode S7, which may have been instrumental in the selection of Method 2(T) in this episode.

The selections which were made of composite "double" methods also indicated the significance of display state for method selection. All six double methods that were selected involved the application of two methods in sequence for which the same system sub-register was active - Method 6(TT) in Episodes D5, D11 and S5, Method 7(TT) in Episode R6D, and Method 8(LL) in Episode S6. It appears that there was a strong inclination to choose a second method which could be implemented using the datasheet that was displayed when the application of the first component method was completed.

7.1.2 The origin and significance of errors

Inspection of Table 7.1 shows that 12 out of the 35 episodes which involved the operation of the program included the commission of between one and five manipulation errors. Episodes in which more than three errors were committed are discussed in this section.

A significant number of errors resulted from the confusion caused by display of "null" graphs corresponding to a set of zero values for the rate of photosynthesis. When Delia selected a T-sheet in Episode D3A the graph displayed defaulted to a rate/C graph for the column corresponding to the minimum L-value, for which all the C-values were zero. This resulted in the display of a null graph, prompting the execution of "panic" operations. Four redundant con_graph(1) operations were executed when the L-value for the selected column was set at zero arbitrary units, and a redundant s_graph(l) operation was executed when the L-value for the selected column was five arbitrary units:

> D: You understand? 30 degrees up there. Right, first of all you do light intensity and connect this [*three redundant con_graph(1) operations executed*] display; it's not connecting. [*s_col(L) operation executed*] Connect. [*redundant con_graph(1) operation executed*] OK. [*m_col(L) operation executed*] Light intensity is five, yeah? Now if we move along, light intensity 10. Let's see what happens, it's not connecting, to rate of photosynthesis, do line graph [*redundant s_graph(l) operation executed*]. Watch, 10 [*m_col(L) operation executed*] is that much, increase light intensity to 15, [*scope_win operation executed*] its that much, 20, 25. [*m_col(L) operation executed*]

Episode D9A provided further examples of errors induced by graph display changes not conforming to user expectations. In this episode Delia explained to Alice how to vary the value of the light intensity by using m_col operations, with the aim of observing changes in a sequence of rate/C graphs:

> D: Now if you go to that arrow here, [*left hand window scroll box*] take it to the end, [*to display minimum L-value column*] now this is light intensity. Connect graph [*redundant con_graph(1) operation*]. Go to display, [*display menu item not selected - the mouse click is off target and an m_col operation is executed*] now connect. [*redundant con_graph(1) operation*] OK go to that. [*second L-col)*] Connect the graph [*redundant con_graph(1) operation*] and just carry on. [*m_col operations executed*] It's connected, it's connected. [*in response to Alice trying to execute a redundant con_graph(1) operation*] What you have to do is just increase, this is light increasing.

There were four instances of the execution of redundant con_graph(1) operations. The first three were in response to instructions from Delia. The first instance occurred after a scope_win command had been executed to display the non-selected minimum L-value column. No change in the graph was seen after the execution of this operation and the currently selected column was now outside the scope of the datasheet window. In the second instance the minimum column was selected, which corresponded to a set of zero C-values, and a null graph was displayed. This led to the execution of con_graph(1) operations intended to "make something happen". In the third instance, even after her instructions resulted in the successful display of a graph Delia still felt the need to ask Alice to execute a con_graph operation. However, in the fourth instance, when Alice tried to execute a con_graph operation of her own accord, Delia realised that her action was redundant and corrected her.

After Tom selected a T-sheet in Episode R3A the C-row position defaulted to the minimum position in which all the cells contained zero values for the rate of photosynthesis. This caused Tom some confusion, as evidenced by the string of redundant con_graph, s_graph, and restore_win(graph_win) operations he executed:

> T: I see. So that's on 30.
>
> S: That's on 30, the optimum temperature.
>
> T: Now we can look at graph, can't we?
>
> R: Yeah. [*{con_graph(2)} and {con_graph(2)} operations executed in succession*]
>
> T: Line graph, or bar graph? [*{s_graph(b)} operation executed*] Think bar graph would be better.
>
> R: We need to change [*{restore(graph window(2)} and {s_graph(b)} operations executed in succession*]
>
> T: Scale?
>
> R: No, level of carbon dioxide.

In Episode R6E Tom was again confused by the non-display of graphs. When he executed an m_sheet(C) operation at the start of Unit-task 3 the column position defaulted to the minimum L-value position for which all the C-values were zero. This resulted in the display of a null rate/T graph, which prompted five errors including the execution of a redundant s_graph operation and a series of redundant window manipulation operations:

> T: OK if we increase carbon dioxide, how can we do that? We can increase it from here, can't we? [*referring to the C-value scroll box and s-sheet(C) operation executed*]
>
> R:Yes. See that's by, um ... [*redundant s_graph(b) operation executed: null graph displayed corresponding to a set of zero rate values*] If you increase carbon dioxide again, [*referring to C-value scroll box*] [*m-sheet(C) operation executed*] keep on increasing. [*m_sheet(C) operation executed to decrease the level of carbon dioxide*] It ...
>
> T: It's not plotting. [*m_sheet(C) operation executed*]
>
> R: No. [*s_graph(l) operation executed*] No, I don't understand that [*series of redundant operations executed: m_win(graph_window(2) / resize_window(graph_window(2) / output(copy) / con_graph(2) / con_graph(2)*] It's connected. [*redundant s_scale(cube) operation executed*] Go up to display and change it to rows. [*s_row(T) operation executed*] That's it,

but we haven't got carbon dioxide at the bottom. [*m_sheet(C) operation executed*] This is one of the problems I found, trying to get the carbon dioxide.

Four errors occured in Episode R6B in which Ruth and Tom were incorrectly trying to change the variables shown on the *vertical* axes of the two displayed graphs. This was an impossible task and they both became increasingly frustrated, resorting to the execution of redundant con-graph operations and arbitrary s_sheet operations:

T: Except now we've got ... OK. Now we can vary this one [*referring to the vertical axis of the rate/L graph displayed in graph window(2)*] because these are both ...

R:That's light intensity [*referring to the vertical axis of the rate/L graph displayed in graph window(2)*]

T: Right, see if we can change this one [*referring to the vertical axis of the rate/L graph displayed in graph window(1)*]to carbon dioxide and that one's light intensity [*referring to the vertical axis of the rate/L graph displayed in graph window(2)*] then we can have a look.

R:Yes, I think this is actually the rate of photosynthesis [*referring to the vertical axis of the rate/L graph displayed in graph window(2)*] rather than ... So what you need to do is put light intensity along here, [*referring to the horizontal axis of the rate/L graph displayed in graph window(2)*] carbon dioxide along the bottom there, [*referring to the horizontal axis of the rate/T graph displayed in graph window(1)*] so be able to compare them in rate of photosynthesis. So if we connect this one. [*sequence of three redundant con_graph(2) operations executed*] Right, it's connected. If we put light intensity along here, [*referring to the horizontal axis of the rate/T graph displayed in graph window(1)*] [*two redundant s_sheet(L) operations executed*] we need to go to temperature, go up to temperature. [*s_sheet(T) operation executed*] So that's light intensity. [*referring to the horizontal axis of the rate/L graph displayed in graph window(2)*] Yeah, if go up to temperature along here and click on arrow up there so we can increase, that's at 40 anyway, that's alright, [*no operation executed*] if you increase level of carbon dioxide, 0.1 anyway [*no operation executed*] if we connect back to this one. [*con_graph(1) operation executed followed by the execution of a redundant con_graph(1) operation*] Right, that's the same now.

The analysis of the episodes with more than three errors indicates that the need for "action confirmation" was a common source of errors. When the execution of an operation did not result in the expected outcome, a frequent response by the user was to seek confirmation of their action by repeating the execution of redundant or arbitrary operations.

7.1.3 Execution of successful methods

An inspection of Table 7.1 shows that 29 episodes featured an attempt to use a successful method. In 18 of these episodes the attempt resulted in an "expert" application of a successful method, as indicated by a zero value for n_r-n_e. The Delia/Alice session featured seven non-expert applications, the Sharon/Uri session featured three such applications, with the remaining non-expert application featured in the Rachel/Tom session. Non-expert applications featured Methods 1, 2, 3 and 6. The episodes in which these non expert applications featured are summarised in Table 7.3.

Table 7.3: Episodes which featured a non-expert application of a successful method

Method	Episode (non-expert application)			
1	D2(∂)	S2(∂)	R4(Δ)	
2	D8(Δ)	S7(∂)	S11(Δ)	
3	D3B(∂)	D7(Δ)	D9A(Δ)	D9B(o)
6	D11(o)			

The method application in all but two of the inexpertly applied successful methods resulted in a change of active system sub-register. Such a change only occurred in four out of the 18 episodes which featured an expert application of a successful method. This implies that a change in active sub-register may be related to non-expert application of successful methods.

7.1.3.1 Non-expert applications of Method 1

Delia used Method 1(C) in Episode D2 to show Alice how to use the program to determine the optimum temperature. After a mistaken initial instruction by Delia to select a T-sheet she instructed Alice to maximise the light intensity in Unit-task 1 by executing an m_sheet(L) operation, and to maximise the carbon dioxide level in Unit-task 2 by executing an m_sheet(C) operation:

> D: OK. Graph isn't connected to data as yet, and first question is, have to use data, says optimum air temperature at maximum rate of photosynthesis, and you have to make sure that light or carbon dioxide are not in limited supply, so basically got to be unlimited. So what you have to do is you go to temperature [*s-sheet (T) operation executed*] and ... No you go to light intensity, [*s-sheet (L) operation executed*] and put it to maximum. [*m-sheet(L) operation executed*]
>
> A: And the same with carbon dioxide.
>
> D: Go to carbon dioxide [*s-sheet(C) operation executed*] and put that up to maximum as well. [*m-sheet(C) operation executed*] [...]

She then advised Alice to inspect the rate/T graph to identify the optimum temperature:

D: OK, right. [*s-graph (l) operation executed*] Just have a look. Display. [*s-row(T) operation executed in error*] If we go down [*scope_win operation executed in an attempt to undo the execution of the previous s_row (T) operation*] to the light intensity there, [*s_col(L) operation executed successfully to undo the execution of the s_row(T) operation*] there you've got maximum carbon dioxide, got maximum light intensity, yeah, and got temperature and rate of photosynthesis. [*referring to the axes shown in the graph_window(1)*] [*m_col(L) operation executed to maximise the value of the light intensity*] Now you just go to connect and connect graph. [*con-graph(1) operation executed*] Just look at the graph, [*inspect the displayed rate/T graph*] and where on the graph do you think is photosynthesis the greatest?

The sub-goal structure of this episode corresponded to Method 1; maximise the value of the light intensity and the level of carbon dioxide and inspect a rate/T graph for these maximum values. However, Delia's demonstration of Method 1(C) involved the execution of four direct manipulations of the display which did not correspond to an expert application. An expert application of Method 1(C) would involve the execution of an m_sheet(C) to maximise the C-value, followed by an m_col(L) operation to maximise the L-value and an inspection of the resulting rate/T graph. As the C-value is maximised by a direct manipulation of the C-sheet, an expert application of Method 1(C) ensures that the C-value and the L-value are both set to a maximum in the system register for the active sheet (in this case the C-sheet). This was not Delia's perception in Episode D2; once the s_sheet(C) operation had been executed the L-value for the current sheet was defined by the L-value for the C-sheet (column position), not the L-value corresponding to the previously selected L-sheet. The fact that the L-value for the L-sheet was set to a maximum in Unit-task 2 was irrelevant, and the relevant L-value could only be a maximum if the L-value for the C-sheet was a maximum. This is illustrated by considering Unit-task 3 in more detail. The action string for this unit-task can be interpreted as the sequence:

1 select a graph type [s_graph(l)]
2 datasheet manipulation error [s_row(T)]
3 datasheet manipulation error correction [scope_win | s_col(L)]
4 goal related error correction [m_col(L)]
5 Graph interpretation [con_graph(1) | inspect_graph (rate/T) | s_graph(b)]

Delia had to execute a "remedial" m_col(L) operation (step 4) to set the L-value to a maximum before inspecting a rate/T graph to identify the optimum temperature. The execution of this operation compensates for the inapplicable m_sheet(L) operation in Unit-task 1.

Despite a non zero value of n_o-n_e in Episode S2, Sharon's execution of Method 1(L) in this episode indicated that she understood the sub-goal structure of this method. In Unit-task 1 she realised that the carbon dioxide level should be maximised and executed an m_row(C) operation on the currently displayed L-sheet to change the C-value from an intermediate value to a maximum value:

> S: OK, [*pointing with the mouse to the L-value scroll box*] just - the light intensity in carbon dioxide should be at maximum so if we have temperature changing. [*pointing to T-columns on the datasheet*] [*max_win(datasheet) operation executed*] Now with light intensity change it here as well. [*redundant s_display and s_view operations executed*] So if want carbon dioxide at a maximum going to be carbon dioxide maximum, [*m_row(C) operation executed*] light intensity changing.

In Unit-task 2 she was confused about her role as a teacher and indulged in a peripheral demonstration of how to manipulate the datacube to change the active datasheet. This led to the execution of arbitrary s_sheet(T) and s_sheet(L) operations:

> S: Temperature going to change. [*undo max_win(datasheet) operation*] Got graph connected to information right now, so this is what we can change. Also can touch this box [*datacube*] and will change whichever bit of information you need [*pointing to the L-sheet face with the mouse, executing an s_sheet(T) operation (by clicking on the T-sheet face), and executing an s_sheet(L) operation (by clicking on the L-sheet face)*] Who's supposed to be answering these questions, both answering? OK. Bad teacher here. Do you want to use that? If we want going to be changing air temperature to find out maximum. Have line with rows. [*pointing to the datasheet rows(C-values)*]

However, when she properly engaged in her teaching role in Unit-task 3 she quickly executed an m_sheet(L) operation to maximise the light intensity; an action consistent with an application of Method 1(L):

> S: OK, light intensity here. [*referring to the L-value scroll box*], clicked so goes to maximum [*m_sheet(L) operation executed*]

She then inspected the displayed rate/T graph in Unit-task 4 to determine the optimum temperature:

> S: OK, so, maximum rate would be the peak in this temperature, 30 degrees. [*referring to currently displayed rate/T graph*]

The C-row had been set in the maximum position for the L-sheet in Unit-task 1, making an m_row(C) operation unnecessary. However, Sharon did not appreciate in Unit-task 3 that the the C-row was already in the maximum position, and she executed a scope_win operation in preparation for executing an m_row(C) operation to maximise the C-value:

> S: [...] Not in limited supplies and carbon dioxide has to go to bottom line, so do you want to click on this one here? [*scope-win operation executed*] OK got that line. [*realising that the maximum row position is already selected*]

The first three unit tasks in Episode R4 constituted an expert application of Method 1(T). When the T-sheet was selected in Unit-task 1 the sheet position defaulted to the optimum temperature location. In Unit-task 2 Ruth advised Tom to execute m_row(C) operations until the level of the carbon dioxide was maximised, and in Unit-task 3 she instructed Tom to inspect the resulting rate/L graph to identify the optimum value for the light intensity. However, in Unit-task 4 she decided that the temperature should be set at the maximum, rather than the optimum, temperature. This led to the execution of a redundant s_sheet(T) operation followed by an m_sheet(T) operation:

> R: Yes, haven't got maximum temperature so ...
>
> T: If I go back here. [*referring to the T-value scroll box*]
>
> R: Yeah.
>
> T: Click that [*redundant s_sheet(T) operation executed*] and increase temperature like that. [*m_sheet(T) operation executed*] Right, I think 40 is the maximum. So this one here ... [*referring to the rate value corresponding to the maximum L-value*]

The completion of this last unit-task indicates a lack of understanding of the difference between the optimum and maximum temperature, rather than a misunderstanding of the sub-goal structure of Method 1.

Delia, Sharon and Ruth appeared to understand the sub-goal structure of Method 1. Despite the non-zero value of n_o-n_e for Episode R4 Ruth also appeared to understand the relationship between the sub-goals and the associated changes in the state of the system. The mismatch between the observed and expert application arose in Episode R4 due a misunderstanding of the task, that is the nature of the optimum temperature; not a misconception of the relationship between the task and the system. The application of Method 1 in Episode S2 did not match an expert application because Sharon indulged in a peripheral demonstration of how to manipulate the datacube. However, Unit-task 3 provides some evidence of a system register related misconception. Sharon was initially unaware that the C-value had been set at a maximum before the execution of the unnecessary s_sheet(T) operation. The execution of the s_sheet(T) operation at the start of Unit-task 3 resulted in the display of a T-sheet with a C-row corresponding to an intermediate position highlighted, indicating that she may have thought that an intermediate C-value was still current after she executed an s_sheet(L) operation to "undo" the s_sheet(T) operation. Stronger evidence for the existence of a misconception related to changes in the active system sub-register was provided by the non-expert application of Method 1(C) by Delia. She initially maximised the L-value by executing an m_sheet(L) operation before selecting a C-sheet, indicating that she thought that maximising the L-value on the L-sheet would also maximise the L-value on the C-sheet; a clear indication that she did not understand that the system behaviour is determined by the values of the *active* system sub-register, that is the values corresponding to the currently selected sheet.

7.1.3.2 Non-expert applications of Method 2

The advice that Delia gave in Episode D8 on the execution of Method 2(T) indicated some confusion about the sub-goal structure of this method. In Unit-task 1 Delia told Alice to maximise the C-value by executing an m_row(C) operation, which implied that, if she was intending to advise the selection of Method 2(T) to explore the effect of changing the temperature, she would instruct Alice to inspect instances of a rate/L graph as the T-sheet location was changed. However, in Unit-task 2 she instructed Alice to execute m_col(L) operations to observe the effect of varying light intensity on the relationship between the rate of photosynthesis and the level of carbon dioxide, as shown by changes in a rate/C graph:

> D: Just go to display and to column, [*s_col(L) operation executed*] go back here, [*scope_win operation executed to display the column corresponding to the minimum L-value*] right, basically go across [*m_col(L) operation executed*] and see how graph increases and decreases. [*interpret-graph rate/C graph*]

She then realised the error in her approach and advised Alice to execute m_sheet(T) operations to observe the effect of changing temperature on a rate/C graph:

> D: [...] Oh! no you've got the temperature at 30, its temperature you're supposed to be .... If you click on there. [*T-value scroll box*]
>
> A: So.
>
> D: Just change that, [*the T-value corresponding to the T-sheet*] only goes up to 40, the temperature. So if you decrease again and just look and see where it increases and where it decreases because increasing temperature. [*explore the effect of changing temperature values by executing m_sheet(T) operations*]

Delia's confused understanding of the method goal structure resulted in an incorrect application of Method 2(T). The m_col(L) operations executed in Unit-task 2 left the L-value at an intermediate value for the T-sheet; thus the instances of rate/C graphs observed as the T-value for the T-sheet was varied corresponded to a non-maximum L-value.

In Episode S7 Sharon provided clear instructions to Uri to apply Method 2(T). Her instructions to Uri clearly indicated that she understood that the C-value needed to be at a maximum and that it was necessary to observe the value of the rate corresponding to the maximum L-value for the rate/L graph as m_sheet(T) operations are executed:

S: Carbon dioxide at maximum, so going to be at bottom line carbon dioxide [*Sharon pointing to the maximum C-value row position*] and last bit of light intensity. [*referring to maximum L-value shown on the rate/L graph*] OK, can just click on it. [*Sharon pointing to the maximum C-value row position*]

Sharon gave rather vague instructions in Unit-task 2 on how to change the temperature by executing m_sheet(T) operations. This led to incorrect execution of s_sheet(C) and m_row(T) operations by Uri. Sharon's clear understanding of how to vary the temperature was evident in her prompt correction of Uri's actions:

S: Yeah, now just have to change the temperature, which you can do on the cube top.

U: Go into that cube there.

S: Yeah, just change temperature [*s_sheet(C) and m_row(T) operations executed in error*] Just press temperature again. Just go back to it. Press temperature. [*s_sheet(T) operation executed*] Start at zero degrees and increase. [*m_sheet(T) operation executed*]

Sharon's execution of Method 2(C) in Episode S11 also indicated a sound grasp of Method 2. The mismatch between an expert application of this method and the observed execution arises from the inclusion of an unnecessary m_col(L) operation at the end of Unit-task 1; probably in response to the display of a null graph.

The execution of Methods 2(T) and 2(C) by Sharon indicated that she understood the relationship between the sub-goal structure and the system state. However, there was further evidence in Episode D8 that Delia did not understand the relationship between direct manipulation of the display and consequent changes in the state of the system. She did not appear to appreciate the effect on the T-sheet sub-register of the m_col(L) operations executed in Unit-task 2, and was unaware that the light intensity was not set at a maximum value in Unit-task 3 as m_sheet(T) operations were executed to explore the effect of varying the temperature. It is interesting to note that at the end of the previous episode, that the L-sheet system sub-register was active with the L-value set at a maximum, implying that she may not have appreciated that the L-value in the active system sub-register had been changed from the maximum value to an intermediate value by the m_col(L) operations she executed.

7.1.3.3 Non-expert applications of Method 3 and Method 6

Episode D3B featured an application of Method 3(T). At the start of this episode the T-sheet was located in the position corresponding to the optimum temperature. However, Delia executed an initial mistaken s_sheet(C) operation:

D: [...] Now if we display carbon dioxide amount. [*s_sheet(C) operation executed*]. Bear with me. Temperature at 30 [*s_sheet(T) operation executed*] and carbon dioxide down here, [*indicating a L-column*] so display columns, [*s_col(L) operation executed in error*] rows [*s_row(C) operation executed to undo the previous s_col(L) operation*] Sorry. Connect. [*con_graph(2) operation executed*] This is ... Get right to the top. [*sequence of m_row(C) and scope_win operations executed to set the C-value at minimum*] Right, at 0.01. [*after executing m_row(C) and scope_win operations to observe the effect of changing the C-value*] Do you want to just talk through the graphs?

A: As increase amount of carbon dioxide the graph seems to, rate of photosynthesis seems to be increasing.

D: Yeah, so that's that question.

It appears that Delia originally thought that the value of the carbon dioxide concentration associated with the T-sheet could be changed by clicking on the C-value scroll box. This led to some confusion which she acknowledged - "Now if we display carbon dioxide. Bear with me." - and she reset the display to show the T-sheet by clicking on the T-value scroll box.

Episode D7 featured an application of Method 3(L). Delia instructed Alice to maximise the carbon dioxide concentration and the light intensity by locating the C-sheet and the L-sheet at the positions corresponding respectively to the maximum C-value and L-value:

D: [...] Find optimum air temperature for maximum rate of photosynthesis.

A: Do I press this button? [*pointing with the mouse to the C-value scroll box*]

D: Well first of all you've got to make sure that neither light nor carbon dioxide are in limited supplies, so go to light and ... yeah, just increase for maximum [*confirming that Alice should execute an s_sheet(C) operation (so as to start to implement a method intended to locate the C-sheet at the maximum C-value position) before attempting to maximise the L-value*].

A: Just press it. [*the C-value scroll box*]

D: Yeah. [*s_sheet(L) operation executed*] So this shows that that's maximum at point one [*0.1*], yeah?

A: And I do the same with this one? [*L-value scroll box*]

D: Light intensity? Yeah. That's maximum as well.

Again there is evidence of a misunderstanding of how values are changed in the system register. The C-value can only be effectively maximised while the L-sheet is selected if an m_row (C) operation is executed to locate the row corresponding to the maximum C-value. As in Episode D2 Delia did not realise that an m_sheet (C) operation that was executed in a previous episode was irrelevant to the current direct manipulation of the L-sheet.

In Episode D9A Delia instructed Alice how to perform Method 3(T). In Unit-task 1 Delia verbally confirmed that the temperature needed to be set at the optimum value. The datasheet was already located in this position but Alice incorrectly selected the C-sheet and executed an m_sheet(C) operation in response to Delia's statement that the light intensity and the level of carbon dioxide needed to be varied:

> D: OK, number two. At this optimum temperature look at the way in which varying the light and carbon dioxide levels affects the rate of photosynthesis. So for this one you've got to have optimum temperature.
>
> A: Which is 30 degrees.
>
> D: Yeah
>
> A: So do I just leave it like this? [*referring to the graph window (1) display*]
>
> D: Yes, leave it at 30 degrees, but got to vary light and carbon dioxide.
>
> A: So got to change it here? [*s_sheet(C) operation executed by clicking on the C-value scroll box and executing an m_sheet(C) operation*]
>
> D: So, if you go back to that [*s_sheet(T) operation executed*], this is set at 30, yeah

It was clear in Episode D9A that Delia understood the sub-goal structure of Method 3(T). Her understanding was emphasised by her correction in Episode D9B of Alice's attempt to execute an illegitimate m_col(L) operation during her explanation of how to vary the value of the carbon dioxide concentration using m_row(C) operations:

> D: Yeah? OK do same for carbon dioxide levels.
>
> A: So I have to change the display first?
>
> D: Yeah.
>
> A: To rows. [*s_row(C) operation executed*]
>
> D: Yeah, good, now you need to go to the top. [*inappropriate scope_win operation executed - left hand horizontal window scroll box clicked on when the top right hand vertical window scroll box should have been clicked on*]
>
> (Researcher: So that should read light intensity at the bottom).
>
> D: OK
>
> A: Now go across? [*illegitimate attempt to execute m_col operation*]
>
> D: No, you need to go to the top, this is carbon dioxide [*illegitimate attempt to execute m_col(L) operation corrected by Delia*] and you have to click up there [*minimum C-value row location*] to get right to the top. Now take it down and see how the graph changes. [*m_row(C) operation executed*] OK just bring it

down [scope_win and m_row(C) operations executed] - so do you notice? [*interpret the changes in the rate/L graph*]

Despite the clear instructions given in Episode D9A and the correction of Alice's actions in Episode D9B, the analysis of Episodes D3A and D7 provides further evidence that Delia misconceived the relationship between directly manipulating the display and the associated changes in the state of the system. The initial mistaken execution of an s_sheet(C) operation in Episode D3B indicates a lack of understanding of which system sub-register was active. In Episode D7, as in Episode D2, the attempt to maximise two variable values corresponding to the same sheet by executing two m_sheet operations indicated a lack of understanding of which operators affect the state of the current system sub-register.

Delia supported Alice in Episode D11 in an application of Method 6(TT), a "double" method consisting of a sequence of two applications of Method 3(T). As in Episodes D9A and D9B, the instructions given to by Delia indicated that she was developing a more accurate appreciation of the relationship between direct manipulation of the display and changes in the system state. She gave clear instructions in Unit-task 1 to execute m_col(L) operations to explore the effect of varying the light intensity, and in Unit-task 2 to execute m_row(C) operations to explore the effect of varying the carbon dioxide level. The mismatch between the expert application of Method 6(TT) and the observed application was simply due a momentary confusion over whether to select rows or columns, and there was no evidence of a misconception concerned with changes of the active system sub-register. The lack of significance of these mistakes is illustrated by the mistake made in Unit-task 1:

D: So if you now change the display.

A: It's columns, no, rows. [*an attempt to execute an s_col operation in error before executing an s_row operation*]

7.1.4 Execution of unsuccessful methods

The application of unsuccessful methods was confined to Episode 6. This episode consisted of a sequence of seven linked confused episodes (Episode 6A - Episode 6G). Various idiosyncratic methods were employed which were based on problem solving with very little evidence of the practice of routine cognitive skill.

Episode R6A was based on an an abortive attempt to use Method 4. Rachel outlined the strategy behind this method in a rather confused fashion in Unit-task 2:

R: What we need is carbon dioxide [*referring to the horizontal axis of the rate/L graph shown in graph window(2)*] along here. [*referring to the horizontal axis of the rate/T graph displayed in graph window(1)*]

T: Along the bottom.

R: And then have light intensity along here. [*referring to the horizontal axis of graph window(1)*] No we have light intensity along here [*referring to the horizontal axis of graph window(2)*] and so we need carbon dioxide along here. [*referring to the horizontal axis of graph window(1)*]

In preparation for a comparison of rate/L and rate/C graphs Unit-task 1 was concerned with configuring the screen to show two graph_windows side by side. At the start of the episode it appeared that only one graph_window was open. However, two graphs were open, with graph_window(2) completely obscured by graph_window(1). It seems that this led Ruth to believe initially that only one graph_window was open as she advised Tom to start a third graph_window. This was opened and placed on top of graph_window(2) by default. Ruth advised Tom to iconise graph window(3) in order to reveal graph_window(2) and to move the location of graph_window(2).

Tom suggested that the temperature should be made the same for each of the displayed graphs. Ruth agreed and an m_sheet(T) operation was executed in an attempt to achieve this sub-goal. This decision indicated a lack of understanding of the relationship between the displayed graphs and the datacube sheets - providing an unconnected graph had at some stage within a unit-task been connected to the active sheet in its current position the graph would correspond to the current sheet value (for example, a T-value) in the sheet sub-register; exercising an m_sheet operation would have no effect in this respect. During Unit-task 3 graph_window(1) was not connected to the T-sheet, and the graphs displayed in this window (pie chart followed by a bar graph) corresponded to the T-value current at the end of Episode R1.

In Unit-task 4 an attempt was made to change the horizontal axis displayed in graph_window(1) to "carbon dioxide". This could have been done by executing an s_col(L) operation. However, Ruth advised Tom to execute a con_graph(1) operation. The result was to produce a rate/L graph and she commented "Now that's the same, but we need to change it". This indicated a misunderstanding of the relationship between a displayed graph and the datacube sheets. Strictly speaking this displayed rate/L graph was not the same - the rate/L graph displayed in graph_window(1) corresponded to a row on a T-sheet, and the graph displayed in graph window(2) corresponded to a row on a C-sheet. She then advised the execution in turn of an s_sheet(C) operation and an s_sheet(L) operation. Given that the aim of Unit-task 3 was to make the temperature the same for both of the displayed graphs, these operations indicate that Ruth did not understand the relationship between direct manipulations of the datacube and the associated changes in which system sub-register was active. In selecting firstly a C-sheet, and secondly an L-sheet, there was no guarantee that the T-value associated with either of these sheets would be the same as the T-value associated with the T-sheet.

Episode R6B was dominated by Tom's confusion of the row values on the vertical axis of the datacube with the values on the vertical axis of a displayed graph:

T: I thought that this Y axis here, [*referring to the vertical axis of the rate/L graph displayed in graph window(2)*] I thought this [*the datacube*] represented carbon dioxide levels, I thought the whole idea of this was to represent some sort of three-dimensional table, so you could have a look at what happened when you vary three different parameters, so I thought that this here was the carbon dioxide, [*referring to the vertical axis of the rate/L graph displayed in graph window(2)*] that was light intensity and this was your temperature varying. [*referring to the value display bar in window(2)*] So I thought this was just a title. [*the vertical axis title*]

Unit-task 1 was concerned with making the horizontal axes of both of the graphs the same in preparation for making the *vertical* axis of one graph carbon dioxide and the *vertical* axis of the other graph light intensity. After first agreeing with Tom's proposal, Ruth corrected his misconception, and advised Tom to execute a sequence of operations which failed to achieve her goal of putting carbon dioxide on one horizontal axis and light intensity on the other horizontal axis. At the end of the episode both windows displayed an instance of a rate/L graph.

Episode 6C featured a second abortive attempt to apply Method 4. In Unit-task 1 an unsuccessful attempt was made to specify carbon dioxide as the variable for the horizontal axis of the graph displayed in graph window(1). There was evident confusion over the relationship between directly manipulating the datacube and changes in the instances of the graph displayed in the connected window:

R: If we connect this graph.

T: Right, if we connect this graph [*referring to graph window(1)*]

R: Yes.

R: And we change to carbon dioxide on the horizontal axis.

T: OK, can do that by going back here. [*s_sheet(C) operation executed*]

R: Yes, no, try temperature [*s_sheet(T) operation executed*]. I can never remember which way round this goes. No maybe it's light intensity [*s_sheet(L) operation executed*]. No that's temperature. [*referring to the horizontal axis of the rate/T graph displayed in graph window(1)*]

A second attempt to make this specification for the graph displayed in graph window(1) was made in Unit-task 2. This time the attempt was successful:

T: Right, want to have carbon dioxide across the bottom?

R: Yes. If we go to display and change it to columns, [*s_col(T) operation executed*] yeah, that's it, so got carbon dioxide along here, [*referring to the horizontal axis of the rate/C graph shown in graph window(1)*] that's basically 0 missed out. [*referring to the initial two zero C-values*]

In Unit-task 3 an attempt was made to make the temperatures associated with each graph the same. As in Episode R6A this resulted in confusion, diverting the attention of both Ruth and Tom from the use of the sought after screen configuration with the two different horizontal graph axes displayed:

> T: I think the temperature is a bit different.
>
> R:That's actually light intensity. [*referring to the value display in graph window(1)*] Need to go to temperature on the arrow. [*referring to the T-value scroll box*]
>
> T: Reduce it?
>
> R:No, increase it to 40.
>
> T: On this one? [*referring to graph window(1)*]
>
> R:Yeah, on that one.
>
> T: Its 50.
>
> R:That's the light intensity.
>
> T: Oh, right.
>
> R: 50 light intensity and 10 is temperature, so if we go back to temperature on arrow up here, [*Tom makes to use the T-value scroll box to decrease the temperature*] on the other arrow [*s_sheet(T) operation executed*] Need to ... yeah that's it, that's at 40.
>
> T: And now we have to ... Shall I do graph, go back to bar graph? [*redundant s_graph(b) operation executed*]
>
> R: And we need to increase light intensity, so if go to light on arrow on bottom, [*s_sheet(L) operation executed*] and take it up to 50 [*m_sheet(L) operation executed; the L-value is initially at 50, Tom decreased the L-value and has to increase it again*] So ...
>
> T: What we've ...
>
> R: So what we've done; go to temperature again on arrow, [*s_sheet(T) operation executed*] and increase that to 40. [*m_sheet(T) operation attempted; T-value already at 40 degrees*] Don't understand why can't have ...

The lack of understanding of the how the display relates to the active system sub-register was evident in Unit-task 3. In Unit-task 2 the rate/C graph displayed in graph window(1) corresponded to an active L-sheet sub-register with the L-value set to a maximum. The rate/C graph displayed in Unit-task 3 corresponded to an active T-sheet sub-register with the L-value in this sub-register set at the minimum value. Apparently equivalent graphs are shown, but they correspond to different sub-registers with different

L-values. The failure to appreciate this difference led to the confusion displayed in Unit-task 3.

Unit-task 4 featured another attempt to specify carbon dioxide as the horizontal axis variable for the graph displayed in graph window(1):

> T: Shall we go back to what we did with columns?
>
> R: Yes, that's display, go to that.
>
> T: Go back to rows? [*s_row(C) operation executed*]
>
> R: Yes. That's right, got the same up here, [*referring to the C-value and T-value shown in the graph window value displays*] just got two different factors. [*on the horizontal axes*]
>
> T: And this is still ... I mean that is temperature [*referring to the T-values shown in the graph window value displays*], yeah. So this is carbon dioxide levels [*referring to the vertical axes of the displayed graphs*], both of these is carbon dioxide.
>
> R: Yeah those two there are rate of photosynthesis, [*referring to the vertical axes of the displayed graphs*] that's carbon dioxide on the bottom, [*referring to the horizontal axis of graph in graph window(1) - this should have been displayed a a rate/L graph, but a display bug showed it as a rate/C graph*] got light intensity along here. [*referring to the horizontal axis of rate/L graph in graph window(2)*]

In Episode R6D Ruth agreed to Tom's suggestion during Unit-task 1 that the level of carbon dioxide should be fixed and that the temperature should be varied:

> R: OK. Right. That reads light intensity. [*referring to the horizontal axis displayed in graph window(1)*]
>
> T: Shall we have a look at varying ... Why don't we keep carbon dioxide fixed then, [*pointing to the C-value displayed in the datasheet values display*] and we can vary temperature, that might work?
>
> R: OK

Unit-tasks 2 and 3 featured an expert application of a truncated version of Method 7(TT). Tom suggested that attention should be focussed on the bar in each of the two displayed bar graphs which corresponded to the same L-value (20 units was chosen as fixed value) and that the location of the T-sheet should be moved to vary the temperature.

> T: We could just have a look at one section of this [*pointing to the datacube*] and say, OK, for something like light intensity 20, [*referring to a specific value of the light intensity on the horizontal axis of the r/L graph displayed in graph window(2)*] so rate of photosynthesis is, what, 20 whatever units you use, [*referring to the corresponding value of the rate on the vertical axis of the r/L graph displayed in graph window(2)*] and now when you vary light intensity for the same section of the graph, [*that is, the bar in each of the two displayed*

bar graphs corresponding to the same L-value (in this case 20 units)] which would be what, it would be this one there wouldn't it, 1,2,3,4; 1, 2, 3, 4. [*identifying the equivalent bar position on each of the two graphs*] It would be this one here, that section there, so if we look at what that section looks like. If you reduce temperature [*m_sheet(T) operation executed to reduce the temperature*] it's getting a lot less, rate of photosynthesis, but by what factor?

As the T-sheet was already selected and the C-row was in the required position, these suggestions corresponded to a fully display-truncated application of Method 3(T). He also suggested that the relative change in the value associated with the chosen light intensity should be observed by comparing the connected rate/L graph shown in graph_window(1) and the unconnected rate/L graph shown graph_window(2). The graph displayed in graph window(2) corresponded to T- and C-values for the T-sheet sub-register set in Episode 6A. These values were the same as those corresponding to the graph currently displayed in graph_window(1) at the start of the episode. The decision to use the graph displayed in graph_window(2) in this way amounted to an application of a fully display-truncated version of Method 1(T). The comparison of the two graphs amounted to a fully display-truncated version of Method 7(TT).

In Unit-task 3 an application of the method was repeated with attention now focussed on one graph only. In this episode, estimates were made of the change in the rate (as represented by the magnitude of the chosen bar in the bar graph) as the T-value was altered by executing m_sheet(T) operations:

T: For a fixed light intensity at 20, and just varying temperature; right it's 25, you increase that to 30, [*m_sheet(T) operation executed to change the temperature to 30 degrees*] then if you connect this one it might be easier to look at that. [*con-graph(2) operation executed and attention now focussed on graph window(2)*]

R: So ...

T: [*m_sheet(T) operation executed to decrease the temperature*] It makes ... [*m_sheet(T) operation executed to increase the temperature*]

R: So it just ...

T: By an increase of 10 degrees, [*m_sheet(T) operation executed to change the T-value from 30 degrees to 20 degrees, followed by the execution of m_sheet operations to change the temperature to 10 degrees and back to 20 degrees*] its only varied by well not that much.

The decision to observe the effect of changing the temperature on a rate/L graph was a change in strategy. Had the problems with manipulating the display in Episodes R6A, R6B and R6C forced this change in strategy; a strategy which was not based on an analysis of the task, but simply on the desire to achieve some interpretable results, even if they were not relevant to the current problem solving activity?

In Episode R6E a fresh attempt was made to consider the relative effects of changing the level of carbon dioxide and the value of the light intensity. Unit-task 1 featured a very confused attempt to establish a rate/C graph in graph window(2):

> T: If you wanted to increase rate of photosynthesis ... Shall we have a look to see what happens if you increase carbon dioxide levels. If you put carbon dioxide across the bottom, [*referring to the horizontal axis of the rate/L graph displayed in graph window(2)*] so if you go to, is it this one here? [*referring to the C-value scroll box*]
>
> R: I think so, yeah. If you click.
>
> T: We need to put this carbon dioxide here [*referring to the left hand C-value edge of the datacube*] don't we. [*referring to the bottom L-value edge of the datacube*] So. [*s_sheet(C) operation executed*]
>
> R: That's light intensity. \ [*referring to the rate/L graph (for a row in a C-sheet) displayed in graph window(2)*] That's what's confusing. [*s_sheet(L) and m_sheet(L) operations executed*] Go up to temperature. [*s_sheet(T) and m_sheet(T) operations executed*] This always confuses me. What I think ... [*s_sheet(C) operation executed*] What I think, if you go over to the edge of this side here, [*referring to the right hand C-value edge of the datacube*] and go in the middle and click, just click on it, [*s_sheet(T) operation executed*] I can never remember how to do this. It's um ... [*s_sheet(C) operation executed*]. Go to display again and change it, [*m_sheet(C) operations executed*] which is up there, [*pointing to the display menu item in the datasheet window*] back to columns. [*s_col(T) operation executed*] That's temperature. [*referring to the rate/T graph displayed (the currently selected column corresponds to a a set of zero rate values - no graph displayed*] [*m_sheet(C) operation executed*] No, go to light intensity, [*s_sheet(L) operation executed*] that's it.

The interaction string for this unit-task highlights Ruth's confusion:

|| s_sheet(C) | inspect_graph(r/L) | s_sheet(L) | inspect_graph(r/T) | m_sheet(L) | s_sheet(T) | inspect_graph(r/L) | m_sheet(T) | s_sheet(C) | s_sheet(T) | inspect_graph(r/L) | s_sheet(C) | m_sheet(C) | m_sheet(C) | s_col(T) | m_sheet(C) | s_sheet(L)

Her confusion was compounded by a number of arbitrary, almost random, m_sheet operations executed by Tom which were not noticed by Ruth. These arbitrary operations are evident in the action string by the absence of associated inspect_graph operations. However, the essential cause of her confusion appears to be her failure to appreciate that the execution of an m_sheet operation would change the currently active sub-register. This led to an inability to form a clear plan for achieving her sub-goal of producing a rate/C graph. At the end of this unit-task a rate/C graph is produced, but the confused nature of the unit-task indicates that Ruth did not appreciate that this graph corresponded to an L-sheet as opposed to a T-sheet. In fact a rate/C graph for an L-sheet could have been produced by simply executing an s_sheet(L) | s_col(T) action string. For a T-sheet a rate/C graph could have been produced by simply executing an s_col(L) operation.

After a discussion in Unit-task 2 to explain the truncated nature of the labelling on the horizontal axis of the graph shown in graph_window(2), Unit-task 3 featured a very confused attempt to explore the effect of varying the level of carbon dioxide. As in Unit-task 1 Ruth was confused about the significance of making another sub-register active:

> R: [...] Go up to display and change it to rows. [*s_row(T) operation executed*] That's it, but we haven't got carbon dioxide at the bottom. [*m_sheet(C) operation executed*] This is one of the problems I found, trying to get the carbon dioxide.
>
> T: If you click ... [*s-sheet(L) operation executed*]
>
> R: If you increase carbon dioxide. If you go over to carbon dioxide and increase that [*referring to the C-value scroll box*]
>
> T: OK, now carbon dioxide is over here, isn't it? [*referring to the vertical axis of the rate/T graph shown in graph window(2)*]
>
> R: No, carbon dioxide is fixed at 0.09 [*referring to the C-value shown in the datasheet value display in graph window(2)*]
>
> T: Right.
>
> R: And this is the rate of photosynthesis up here. [*referring to the rate axis of the rate/C graph shown in graph window(2)*]
>
> T: OK, lets have a look at it this way. If we reduce it. [*referring to the C-value*] [*s_sheet(C) operation executed followed by an m_sheet(C) operation*]
>
> R: Right, at the moment it is at 0.8.
>
> T: Right, its decreasing now, [*m_sheet(C) operation executed to decrease the C-value*] decreasing carbon dioxide from 0.07. Go to display choose columns. [*s_col(L) operation executed*]
>
> R: Right that's temperature. [*referring to the rate/T graph shown in graph window(2)*]
>
> T: [*s_sheet(L) operation executed*] Carbon dioxide there. [*referring to horizontal axis of the rate/C graph shown in graph window(2)*] Shall we try varying it again? [*referring to the C-value scroll box*

Unit-task 4 again demonstrated that Ruth did not understand the significance of making another sub-register active. She advised Tom to maximise the temperature by selecting a T-sheet and executing an m_sheet(T) operation to maximise the T-value. When an L-sheet was selected again, to display a rate/C) graph, Ruth became confused because the temperature was "still" at 10 degrees, that is the value corresponding to the L-sheet sub-register, not the T-sheet sub-register:

R: Basically this is ... If we look at this along here. [*referring to the horizontal axis of the rate/C graph displayed in graph window(2)*] If we increase temperature to its maximum. So go to temperature and increase it. [*s_sheet(T) and m_sheet(T) operations executed*] Right go to light intensity and increase that [*s_sheet(L) operation executed*] on the arrow, yes. I don't understand why the temperature is decreasing. [*m_sheet(L) operation executed to maximise the L-value*]

Ruth and Tom unsuccessfully applied another idiosyncratic method in Episode R6F This method involved a comparison of a rate/C graph corresponding to a column selected on an L-sheet and a rate/T graph corresponding to a column selected on a C-sheet. The method was based on the idea of comparing these two graph instances with the same C- and L-values considered for each graph. An attempt to focus on the same C-value in each graph was made by maximising the C-value in graph_window(2) by executing an m_sheet(C) operation, and focussing attention on the rate value corresponding to the maximum C-value shown on the horizontal axis of the rate/C graph displayed in graph_window(2):

T: Hold on, need to fix this as well. [*referring to the C-value shown in the graph window(1) values display*] This is the carbon dioxide level. Let's have a look at it for 0.1.

R: OK, so go to carbon dioxide and increase it. [*m_sheet(C) operation executed which first decreases the C-value to the minimum value and than increases it to the maximum value*] Right.

T: 0.1 is this value here, isn't it? [*referring to the maximum C-value shown on the horizontal axis of the rate/C graph displayed in graph window(2)*] Last one. So the last one corresponds to about 10 [*referring to the rate value corresponding to the maximum C-value on the horizontal axis of the rate/C graph shown graph window(2)*]. Is this value here the temperature? [*pointing to the L-value shown in the datasheet values display in graph window(1)*]

The L-value associated with graph_window(2) was the L-value in the L-sheet sub-register which was currently at the maximum value. An attempt to focus on the same L-value in each graph was made by maximising the L-value in graph_window(1) by executing an m_col(L) command until the L-column corresponding to the maximum L-value was highlighted:

R: No that's light intensity. You could increase light intensity. If you go all the way along here, you click on that. [*m_col(L) operation executed*]

T: So light intensity here is 10? [*referring to T-value shown in the values display for graph window(2)*]

R: No that's the temperature

T: Yeah.

R: If you go to the arrow there at bottom in the corner [*referring to the right hand horizontal window scroll box*], click on it [*scroll_win operation executed*], keep on clicking, [*scroll_win operation executed*] now click on this column here. [*referring to the L-column corresponding to the maximum L-value*] [*m_col(L) operation executed*] If that's going to be 0.1, [*referring to the maximum L-value shown on the horizontal axis of the rate/L graph displayed in graph window(2)*] light intensity is at 50 [*pointing to both graphs*], and got temperature and carbon dioxide. [*referring to the horizontal axis variable for both graph window(1) and graph window(2) respectively*]

T: This is for 50 as well? [*referring to the L-value shown in the values display for graph window(2)*]

R: Yeah, that's 50. Everything is same. Looking at this, [*referring the portion of the rate/C graph corresponding to the maximum L-value in graph window(2)*] is 0.1. [*referring to the C-value shown in the values display for graph window(2)*] So ...

T: Um ...

It appears that Ruth and Tom were trying to keep two variables constant (light intensity and carbon dioxide concentration) but they did not make a clear attempt to vary the third variable (temperature). However, the last unit-task in the previous episode was concerned with the effects of varying the temperature. Ruth and Tom appeared to have "lost their way" in an attempt to adopt the classic "only vary one thing at a time" paradigm.

The episode concluded with a discussion in Unit-task 4 based on a numerical estimate of which factor to increase. Tom stated that:

T: Just look at the factors, [*con_graph(1) operation executed*] increasing carbon dioxide by what, 0.01 each time, and yet that is having effect on.

Tom commented that he did not think light intensity and temperature were "as good" as carbon dioxide, that is, changes in value of these two variables did not have as significant an effect as changes in the value of carbon dioxide. Ruth agreed with this assertion and an m_sheet(C) operation was executed to confirm this conclusion. At the end of this long and confused episode Ruth and Tom were still unable to achieve their original aim of comparing a rate/L graph with a rate/C graph!

Episode 6 provided an extensive and convincing illustration of the same lack of understanding by Ruth and Tom as that shown by Delia of the relationship between the direct manipulation of the display and the associated changes in the active system sub-register. Despite many attempts, this lack of understanding prevented them from making a successful attempt to answer Question 5.

7.2 Interaction with *Bioview* from a task perspective

Each episode was focused on a specific task - an initial illustration of the direct manipulation of the datacube or a datasheet, a demonstration by a teacher-student of how to use *Bioview* to answer one of the set questions, an attempt by a student to use *Bioview* to answer one of the set questions with guidance from a teacher-student, or a final reflection on the merits of *Bioview*. The episodes and method selections which corresponded to the use of *Bioview* to answer the set questions are shown in Table 7.4.

Table 7.4: Episodes and method selections corresponding to each question

Task	Session					
	Delia/Alice		Sharon/Uri		Ruth/Tom	
	Demo.	Advice	Demo.	Advice	Demo.	Advice
Illustration	D1[*Idio*]		S1[*Idio*]		R1[*Idio*]	
Question 1	D2[*1(C)*]	D7[*3(L)*] D8[*2(T)*]	S2[*1(L)*]	S7[*2(T)*]		R2[*1(L)*]
Question 2	D3A[*3(T)*] D3B[*3(T)*]	D9A[*3(T)*] D9B[*3(T)*]	S3A[*3(T)*] S3B[*3(T)*]	S8A[*3(T)*] S8B[*3(T)*]		R3A[*3(T)*] R3B[*3(L)*]
Question 3	D4A[*1(T)*] D4B[*3(T)*]	D10[*3(T)*]	S4[*3(T)*]	S9A[*1(T)*] S9B[*3(T)*]		R4[*1(T)*]
Question 4	D5[*6(TT)*]	D11[*6(TT)*]	S5[*6(TT)*]	S10[*Recall*]		R5[*Recall*]
Question 5	D6[*Recall*]	D12[*Recall*]	S6[*8(LL)*]	S11[*2(C)*]		R6A{*4*} R6B{*4*} R6C{*4*} R6D[*7(TT)*] R6E{*Idio*} R6F{*Idio*} R6G{*Idio*}
Discussion						R7[*Idio*]

There were some marked similarities in the task profiles of the three sessions. All the sessions included an initial episode in which the teacher-student demonstrated how to manipulate the datacube and datasheet widgets. Each session included a sequence of episodes in which the student used *Bioview* to answer the set questions in turn with guidance from the teacher-student. Prior to this sequence the Delia/Alice and Sharon/Uri sessions included a sequence of episodes in which the teacher-student demonstrated the use of *Bioview* to answer each question. The final episode in the Ruth/Tom session consisted of a discussion of the design features of *Bioview*.

An inspection of Table 7.4 shows that the there was a close match between the methods selected for each question and the set of possible successful methods identified in Section 6.2.3, with the application of a successful method featured in 29 out of the 39 episodes related related to a question. Three of the remaining episodes featured aborted attempts to apply a successful method, and four episodes were based on recall of experience of previous episodes. Only three episodes related to a question featured the application of idiosyncratic methods. This close match implies some correlation between learner and designer models of the task-system relationship.

In the following sections, the episodes concerned with the use of *Bioview* to answer the set questions are considered from a task perspective. The task goals implicit in each question are analysed in terms of the sub-goal structures of the methods applied in each episode, with the presence or absence of task related truncation used as an indicator of the the sub-goal structure that the learner was adopting. The learners' preferred use of direct manipulation techniques is also analysed with respect to the use of direct manipulation to fix or vary variables in answering each of the set questions. These analyses provide an in-depth consideration of the relationship between learner and designer models of the task-system relationship.

7.2.1 Using *Bioview* to answer Question 1

In Question 1 the learners were asked to find the optimum air temperature for the maximum rate of photosynthesis. They were instructed to make sure that neither light nor carbon dioxide were in limited supplies.

Method 1 was employed by Delia and Sharon when they demonstrated how to use the program (Episodes D2 and S2), and by Tom when he used the program with Ruth's help (Episode R2). Both Delia and Sharon recommended the use of different methods when they supervised the use *Bioview*. Sharon provided instructions for an application of Method 2 (Episode S7). Alice attempted to answer this question twice with help from

Delia. The first attempt (Episode D7) featured an application of Method 3 and the second attempt (Episode D8) featured an application of Method 2.

The sub-goals of making the values of the light intensity and the level of carbon dioxide non-limiting were simplistically interpreted in all the sessions as requirements to maximise the level of carbon dioxide and the value of the light intensity. For example, in Episode R2 the L-sheet corresponding to the maximum L-value and a row corresponding to an intermediate C-value were already selected at the start of the episode. Ruth informed Tom that the light intensity was set at a maximum and requested him to select the row corresponding to the maximum C-value. In response to his question as to why this is necessary she explained that the selection of this row gave the maximum value of carbon dioxide:

> R: So at the moment got maximum light intensity, which is at 50 and carbon dioxide. We need to, if you go on [*scope_win operation executed*] ... and pull it down, [*referring to a C-row*] should go down, if keep on going down. Go to 0.1 [*referring to the maximum value of carbon dioxide*] and click on it.
>
> T: Why?
>
> R: Because if you do that, it gives you the maximum amount of carbon dioxide. So if you click on here, [*m-row(C) operation executed*] it gives you unlimited supplies of light intensity and carbon dioxide.

There was no convincing evidence that either the teacher-students or the students appreciated the concept of a limiting factor. Tom indicated that he might have some idea of this concept when he enquired in Episode R2 whether making sure that neither light or carbon dioxide are in unlimited supplies meant that they were "unfixed". Ruth did not share his interpretation, and replied that "it means basically at its maximum - it's not at its minimum". However, in all three of the sessions there was evidence of a correct understanding of the concept of an optimum temperature. In Episode S2 Sharon stated that the "maximum rate should be the peak in this temperature - 30 degrees". Delia asked Alice in Episode D2 "where on the [rate/T] graph do you think photosynthesis is the greatest?" Alice immediately replied "30 degrees". In Episode R2 Ruth informed Tom that "if you look at the [rate/T] graph now you can see that it's 30". Tom replied "30, yes, 30 degrees".

An inspection of Table 7.1 reveals that the application of Method 1(C) in Episode D2, Method 1(L) in Episodes R2 and S2, and Method 2(T) in Episode S7 were not task-truncated. However, the applications of Method 3(L) and 2(T) in Episodes D7 and D8 respectively were task-truncated. In both of these applications the value of the light intensity was fixed using a direct manipulation technique; an m_sheet(L) operation was executed in Episode D7 and an m_col(L) operation was executed in Episode D8. The failure to perform the appropriate inspect_graph(r/C) operations to identify a fixed level of carbon dioxide was probably due to Delia's misconception of the relationship between direct manipulation of the display and changes of the active system sub-register. At the start of Unit-task 1 in Episode D7 she executed an s_sheet(C) operation which defaulted the C-sheet location to the maximum C-value position. It is likely that Delia assumed that this maximum C-value was current throughout Episodes D7 and D8, despite the fact she was manipulating in turn an L-sheet and a T-sheet during these episodes.

The instructions in Question 1 emphasised the need to consider changes in the rate of photosynthesis in terms of the interaction between all three variables. In order to ensure that the rate was not limited by either the light intensity or the level of carbon dioxide the user was instructed to make sure that neither light or carbon dioxide were in limited supplies. These instructions implied that both the light intensity and the level of carbon dioxide should be fixed at a non-limiting value. The request to find the optimum temperature implied that the temperature should be varied to find the temperature at which the rate of photosynthesis was a maximum. Thus an appropriate set of sub-goals for the application of a method to answer this question was (i) set the value of the light intensity at a non-limiting value, (ii) set the level of the carbon dioxide to a non-limiting value, and (iii) vary the temperature to identify the optimum value for the temperature. Given the learners' simplistic assumption that non-limiting values were equivalent to maximum values this set was interpreted by the users in a modified form as (i) set the value of the light intensity at a maximum value, (ii) set the level of the carbon dioxide to a maximum value, and (iii) vary the temperature to identify the optimum value for the temperature.

The need to explicitly consider the values of all three variables indicates that the applications of successful methods would not be truncated when applied to answer Question 1. The lack of task related truncation in the three applications of Method 1 and the single application of Method 2(T) indicates that the learners were considering the values of all three variables. If the interaction history prior to Episode D7 is considered, the execution of the task-truncated methods in this episode and Episode D8 can also be interpreted as indicating that the learners were considering the values of all three variables.

In Episodes D2, S2 and R2 Method 1 was applied to directly manipulate the system display to fix the values of two of the variables. In Episodes D7 and D8 the omission of an inspect_graph(r/C) operation, that is, the operation which did not involve direct manipulation, led to the execution of direct manipulation operations to fix the value of one variable (light intensity) and to vary the value of one variable (temperature). This combination of using direct manipulation to fix one variable and vary one of the other variables also occurred in Episode S7. However, the version of Method 2(T) applied in this episode was not task-truncated, so the reason for choosing this combination was not due to task-truncation. The fact that the method was display truncated also did not explain Sharon's decision to recommend a method in which only one variable was fixed by direct manipulation. The display truncation arose from the omission of an m_row(C) operation, not an s_sheet(T) operation. It was not the case that an active T-sheet at the start of the episode prompted her to recommend a method based on the direct manipulation of the T-sheet.

If direct manipulation is an intuitive human-computer interaction paradigm it seems reasonable to assume that Method 1, the only successful method in which the values of two of the interacting variables are fixed by direct manipulation, would be a popular choice as a method applicable to Question 1. In addition, an application of this method results in a static rate/T, which would enable the user to identify the optimum temperature by simply inspecting the graph; a conventional approach which most students would be familiar with from work in other contexts. In fact three of the six episodes associated with Question 1 (Episode D2, Episode S2 and Episode R2) featured an application of Method 1. Each of these episodes was concerned with the first attempt in each session to answer this question, indicating that the decision to adopt this method had not been influenced by the interaction history of the session. The conclusion is supported by the lack of display-related truncation in Episodes D2 and S2.

The use of task-truncated forms of of Methods 3(L) and 2(T) in respectively Episodes D7 and D8 makes it difficult to draw any conclusions from the choice of these methods about the preferential use of direct manipulation techniques. However, the choice of Method 1 in the initial episode of each session, indicated that the conventional inspection of a static graph was preferred to the observation of a dynamic graph generated by manipulating the datacube or a datasheet. It seems that the specific instructions to set the light intensity and the level of carbon dioxide at non-limiting values encouraged the users' to employ direct manipulation techniques to fix two variable values.

7.2.2 Using *Bioview* to answer Question 2

In Question 2 the learners were asked to look at the way in which varying the light and carbon dioxide levels at the optimum temperature affected the rate of photosynthesis. There were no explicit instructions to fix the value of the third variable, that is, fix the value of the light intensity if the level of carbon dioxide was varied or fix the level of carbon dioxide if the value of the light intensity was varied.

Method 3 was used in all ten episodes in which an attempt was made to answer Question 2 (Episodes D3A, D3B, D9A, D9B, S3A, S3B, S8A, S8B, R3A, and R3B). Method 3(L) was employed in Episode R3B, and Method 3(T) was applied in the other nine of these episodes. All ten applications of Method 3 were task-truncated.

The interpretations of the dynamic sequences of graphs produced by an application of these methods were made almost entirely in terms of the relative direction of the change in the rate and the change in the level of carbon dioxide or the light intensity. This was illustrated by the comments made by the learners as they interpreted the graph sequences. The following comments were made as changes in the instances of rate/C graphs were observed:

- So overall generally as you increase light intensity rate of photosynthesis increases as well. [*statement made by Delia in Unit-task D3A/3*]
- Seems to be increasing. [*statement made by Alice in Unit-task D9/2 in response to a question from Delia*]
- Doesn't seem to be changing very much after 40 degrees. [*statement made by Sharon in Unit-task S3/3*]
- It's also increasing, and then sort of stops but really increasing sharply. [*statement made by Sharon in Unit-task S8/2*]

And comments prompted by interpretation of instances of rate/L graphs were:

- As increase amount of carbon dioxide the graph seems to, rate of photosynthesis seems to be increasing. [*statement made by Alice in Unit-task 3B/1 which was confirmed by Delia*]
- Seems to be increasing as you increase amount of carbon dioxide. [*statement made by Alice in Unit-task 9/3 in response to a question from Delia*]
- Yeah, each time see it will be increasing each time. [*statement made by Sharon in Unit-task S3/2*]
- So it's increasing. [*Statement made by Sharon in Unit-task S8/1*]
- So as you increase carbon dioxide the rate increases as well. [*statement made by Uri in Unit-task S8/1 in response to Sharon's statement*]
- Basically intensity increasing, so could say the light intensity is sort of like proportional to carbon dioxide level. [*statement made by Tom in Unit-task R3A/2*]

With the possible exception of the comments made by Sharon in Episodes S3 and S8 and Tom in Episode R3A the learners appear to have simply perceived that as either the C-value or the L-value was increased the rate of photosynthesis increased. This was evident when Sharon and Uri joined voice in Episode S8 to state that "as you increase light intensity and carbon dioxide level the rate is also increasing". Tom's statement in Episode R3B, in which Method 3(L) was applied, that "as you increase carbon dioxide level here, temperature increases and the rate of photosynthesis increases" also indicated this rather restricted interpretation by the learners. It is clear that the form of the displayed graphs were not interpreted in terms of an approach to a non-limited state.

The task-truncated form of all the applications of this method indicates that the value of only one variable was fixed when Question 2 was attempted. This universal task-truncation was consistent with the task prescription given in Question 2. Explicit instructions were only given to fix the value of one variable (temperature). The instruction to fix the temperature without explicit mention of fixing the value of a second variable implied that students would ignore the effect of the value of carbon dioxide if the light intensity was varied and vica versa. This implication is confirmed by an inspection of the relevant Episode Record sheets. For example in Episode S8A the light intensity is only referred to in general terms - "Light intensity is just at the bottom, isn't it" - whilst it is explicitly stated by Sharon that the temperature is constant. Fixing the value of only one variable led in turn to the application of task truncated successful methods.

The exclusive selection of Method 3 indicates that direct manipulation techniques were universally applied to explore the effects of varying the light intensity and the level of carbon dioxide. An inspection of the relevant Episode Record sheets confirms this indication. For example, in Episode S8 the effect of varying the level of carbon dioxide was explored by directly manipulating the T-sheet. A series of m_row(C) operations were executed and the resulting instances of rate/L graphs were inspected:

> S: You've got to see way in which varying light and carbon dioxide levels affects the rate. [*m_row(C) operation executed to locate an intermediate row position*]
>
> U: That would be less.
>
> S: Yeah, so if you've got 30 degrees, so if carbon dioxide level, the light intensity levels all varying, how it affects the rate. So if you go back to the top, this button here [*scope_win operation executed to display the row corresponding to the minimum position*] and sort of carry on down. [*m_row(C) operation executed*]
>
> U: What, this is the carbon dioxide? [*referring to the C-values displayed for each row shown in the datasheet window*]
>
> S: Yeah, so you're varying the carbon dioxide levels. Light intensity is just at the bottom, isn't it. [*referring to horizontal axis of the rate/L graph displayed in graph window(1)*] You've got 30 at degrees and the temperature is constant.
>
> U: Yeah
>
> S: Well you're just changing carbon dioxide levels.
>
> U: So go down. [*scope_win operation executed*]
>
> S: So it's increasing.
>
> U: So as you increase carbon dioxide the rate increases as well. Right. [*m_row(C) operation executed*] [*scope_win operation executed*] So this is the peak, yeah? [*Uri pointing to the row corresponding to the maximum C-value*]

The observed method selections were consistent with the rather vague task prescription in Question 2. The students were simply asked to "look at the way in which varying the light and carbon dioxide levels affects the rate of photosynthesis". This contrasted strongly with the well defined task in Question 1 of finding a specific value for one of the variables. The request to explore the effects of varying the levels of light and carbon dioxide rather than determine a specific value for one of these variables implied that the use of a direct manipulation technique to produce an animated sequence of rate/variable graphs would be appropriate. This in turn implied that an application of either Method 2 or Method 3 would be appropriate. The exclusive use of Method 3 confirmed this implication.

7.2.3 Using *Bioview* to answer Question 3

In Question 3 the learners were simply requested to try to find the optimum level of either light or carbon dioxide. No instructions were given to set the temperature or the light intensity to a fixed value if the optimum level of carbon dioxide was being determined. Likewise no instructions were given to fix the value of the temperature or the level of carbon dioxide if the optimum value of the light intensity was being determined.

Seven episodes featured attempts to answer Question 3. Method 1(T) was applied in three episodes (Episodes D4A, S9B, and R4). None of these applications of Method 1 were task-truncated. However, the applications of this method in Episodes D4A and S9A were fully display-truncated, leaving no opportunity for task related truncation to take place. In Episode R4 both the level of the carbon dioxide and the value of the temperature were deliberately fixed. Method 3 (T) was applied in the other four episodes (Episodes D4B, D10, S4, and S9B). All of the applications of Method 3 were task-truncated.

In all the episodes the idea of an optimum value for light intensity or carbon dioxide concentration was interpreted as the value at which the maximum rate of photosynthesis occurred; there was no apparent use of the idea of the light intensity or the level of carbon dioxide reaching a non-limiting value. In Episode S4 Sharon described the optimum level of the light intensity as "where it stops changing, or starts to decrease", and in Episode S9A she described it as "where it stops changing, where it stops increasing and starts decreasing, or just stops increasing and remaining the same". The optimum value for the light intensity was also interpreted in Episode R4 as the value at which the maximum rate of photosynthesis occurred, with the value being determined by inspecting a rate/L graph. The same interpretation was evident in Episode D10 when Delia advised Alice to determine the optimum value of the level of carbon dioxide by inspecting a rate/C graph to determine where the rate was the greatest:

D: And it says you've got to now find an optimum level of either light or carbon dioxide, so we decided to do light, so.

A: Change display first to columns. [*s_col(L) operation executed*]

D: Yeah, OK, increase the [*light intensity*]. Just see where rate of photosynthesis is the greatest. [*execute m_col and scope_win operations and inspect the rate/C graphs displayed*]

A: 50 degrees.

D: Yeah, and 50 light intensity, [*Delia corrects Alice's incorrect use of units*] yeah?

A: Yeah.

D: So that would be optimum, do you know why?

A: Because at point where makes most photosynthesis.

D: Yeah, where rate is greatest. [...]

The lack of specific instructions to fix the value of the temperature and either the light intensity or the level of carbon dioxide implies that method applications to answer Question 3 might well be doubly task-truncated; once in respect of not fixing the temperature, and again in respect of not fixing the level of light or carbon dioxide.

A consideration of the application of the fully display-truncated applications of Method 1(T) cannot provide an insight into the users' decision on whether or not to fix the values of one or two variables - task truncation was impossible in the fully display-truncated versions of Method 1. The lack of task related truncation in Episode R4 was consistent with deliberate fixing of the values of the temperature and the level of carbon dioxide. In addition to being task-truncated the applications of Method 3(T) in Episodes D4B, D10, S4, and S9B were also display truncated. Inspection of Table 7.2 shows that in particular the s_sheet(T) and m_sheet(T) operations were not executed in these applications. Hence these applications could only be "passively" task truncated with respect to fixing the value of the temperature. However, task truncation with respect to fixing the value of the light intensity or the level of carbon dioxide did occur in all four episodes, indicating that the minimal task description in Question 3 resulted in a failure to consider the effects of the current values of all three variables.

The task in Question 3 was similar to the task in Question 1 - identify a specific value, that is, the optimum value, of a variable. The method selection in the episodes related to Question 1 indicated that the opportunity to determine a specific value by inspection of a static graph made Method 1 a likely choice for such tasks. The method selections in the episodes corresponding to attempts to answer Question 3 only provide limited support for this indication. While three of the episodes did feature an application of Method 1, the other four episodes featured an application of Method 3.

7.2.4 Using *Bioview* to answer Question 4

In Question 4 the learners were asked if there was any difference in the effect of increasing the levels of light or carbon dioxide on the rate of photosynthesis. The task in this question is an extension of the tasks in Question 2 - instead of independently exploring the effects of varying the levels of carbon dioxide and light intensity, the effects of these variations are now compared. Unlike Question 2 there were no instructions in this question to set the temperature at its optimum value.

Two of the five episodes concerned with answering this question (Episodes S10 and R5) consisted of a discussion of the question based on recall of the experience gained in previous episodes. The discussion in both of these episodes was very limited, without a clear attempt being made to answer the question. The remaining three episodes (Episodes D5, S5 and D11) featured an application of Method 6(TT). The applications of Method 6 were all task truncated by the omission in each application of two inspect_graph operations. In addition all three applications were fully display-truncated, with the first Method 3(T) component in Episode S5 completely replaced by recall of previous experience.

The concept of limiting factors does not appear to be used to explain the difference in form between the displayed rate/C and rate/L graphs. For example, in Episode S5 Uri said that there was "only a slight difference" between the effects. Sharon replied, with reference to the variation of the rate with light intensity, that "it goes up to a point". After m_row(C) operations had been executed in Episode S5 to explore the effect of changing the value of the carbon dioxide concentration Sharon stated that "it looks like you can increase carbon dioxide and the rate will increase, but with light comes to a point where it is optimum and stops increasing". Sharon appreciated that the form of the instances of the two types of graph were different, but she did not infer from the differences that the instances of the rate/L graphs demonstrated the value of the light intensity approaching a non-limiting value. The attempts to answer the question in Episodes D5 and D11 simply resulted in a comparison of the way in which the rate changed when the C-value and the L-value were changed without reference to the fact that the form of instances of the two types of graph were different. For example, in Episode D11 Alice observed that "light intensity seems to be increasing slower than carbon dioxide" ; a conclusion which was supported by Delia:

> D: Yeah, because this starts increasing from the origin [*referring to the rate/L graph displayed in graph_window(2)*]. This doesn't start until it's there. [*referring to the rate/C graph which is displayed in graph_window(1), pointing out that the first two C-values are zero*]

The lack of specific instructions to fix the variables that were not being varied implies that there was a good chance that the applications of Episode 6(TT) would be task truncated. As all the applications of this method are fully display truncated, task-truncation is only possible through the omission of inspect_graph operations These operations were omitted, indicating that the users were exploring the effect of changing the value of one variable without considering the value of the other variables.

Given that the task in Question 4 is an extension of the tasks in Question 2, it is reasonable to expect the method selections in the episodes relating to each question would be similar. This was the case; Method 3(T) was universally selected in the methods related to Question 2, and Method 6(TT), a double method in which Method 3(T) is applied twice in sequence, was universally selected in the episodes related to Question 4. As in Question 2 the request to explore the effects of varying the levels of light and carbon dioxide rather than determine a specific value for one of these variables encouraged the use of a direct manipulation technique to produce an animated sequence of rate/variable graphs.

7.2.5 Using *Bioview* to answer Question 5

In Question 5 the learners were asked which factor they would try to increase more to attempt to increase the rate of photosynthesis further at the fastest rate shown in the data. Episodes D6, D12, S6, S11 and R6 were associated with attempts to answer this question. Episodes D6 and D11 were based on the recall of previous experience. Episode S6 featured a full task and display truncated version of Method 8(LL) and Episode S11 featured a task truncated version of Episode 2(C).

None of the episodes included a clear indication that the learners either understood the concept of a limiting factor or that they were able to use this concept to explain the relationship between the three principal interacting variables in photosynthesis.

In Episodes D6 and D12 the learners proposed to simply compare the rates of photosynthesis for rate/C and rate/L graphs. Delia agreed with Alice's suggestion in Episode D6 that "you could have a look at both carbon dioxide and light intensity and see where the rate of increase of photosynthesis is the most, and see which one affects the photosynthesis most". In Episode D12 Alice suggested that the light intensity should be increased further. This suggestion was corrected by Delia with reference to the displayed rate/C graph when she replied that Alice should "read off scale and see which one seems more". Alice agreed to this suggestion and Delia concluded that "basically it's factor which causes the biggest change in the rate of photosynthesis which was carbon dioxide".

In response to Uri's incorrect answer to the question in Episode S6 Sharon indicated that she was aware that the form of the rate/L graph indicated that the increase in the rate of photosynthesis was approaching zero, but she did not explicitly state that this was an approach to a non-limited situation:

> U: Light, I guess.
>
> S: No, it's stopped doing it. [*referring to how the rate of photosynthesis is changing with light intensity*] (mumbles) Increase wasn't that big of carbon dioxide, keeps increasing doesn't it.
>
> U: What, when I was increasing this? [*pointing to the C-value rows on the displayed T-sheet (actually in Unit-task S5/2 Uri was changing the C-value by decreasing it)*]
>
> S: When increase carbon dioxide levels, the levels, the rate also increasing.
>
> U: What, so the more carbon dioxide the faster ...

In Episode S11 Uri correctly answered that the level of carbon dioxide should be increased. However, his answer was rather tentative, and Sharon used Method 2(C) to demonstrate to him the effect of changing the level of carbon dioxide. Again there was no evident use of the concept of limiting factors.

The use of the program in Episode R6 was convoluted and confused, as described in Section 7.1.4. Problems with manipulating the program prevented Ruth and Tom from addressing the question properly. In desperation they attempted to apply a number of idiosyncratic methods which were typically based on the "only change one variable at a time" paradigm. The episode was concluded with an assertion that the level of carbon dioxide should be increased further, but this assertion did not involve any apparent use of the idea that the C-value had not reached a non-limiting value:

> T: [...] but I don't think either of those two [*that is, light intensity and temperature*] are as good as carbon dioxide.
>
> R: Yeah, carbon dioxide.
>
> T: Just look at the factors, [*con_graph(1) operation executed*] increasing carbon dioxide by what, 0.01 each time and yet that is having effect on ...
>
> R: So by increasing carbon dioxide rate [*s_sheet(C) operation executed*] that would be the factor to increase.
>
> T: Yes, as it goes up like that, [*m_sheet(C) operation executed to increase the level of carbon dioxide*] still having an effect. [*m_sheet(C) operation executed top decrease the level of carbon dioxide*] So yeah, [*m_sheet(C) operation executed to increase the level of carbon dioxide*] carbon dioxide.

7.3 Application of the GOMS methodology

The GOMS analysis of the use of *Bioview* with the PSYNTH datacube described in Chapter 4 has been used as a framework for the analysis of the human-computer interaction observed in the laboratory sessions. It is possible to represent each session as a sequence of task-related episodes, with each episode consisting of a small number of unit-tasks. These unit-tasks have been expressed in terms of the operations defined in Section 4.3.1, and most of the episodes featured successful methods corresponding to those identified in Section 6.2.3. In the following sections the applicability of adopting the GOMS approach as a basic framework for representing the human-computer interaction associated in the three observed sessions is critically assessed with respect to the defining features of GOMS - the representation of human-computer interaction in terms of hierarchical goals and sub-goals, the identification of successful methods for achieving these goals, and rule based method selection.

7.3.1 Goal structure

With the exception of the initial demonstration episode in each of the sessions and the final reflective episode in the Ruth/Tom session, the episodes were characterised by goal directed behaviour. It was possible to divide each episode associated with using the software to answer a question into a small number of unit-tasks with each unit-task associated with a well defined sub-goal, such as maximising the value of a variable or inspecting a graph. Table 7.5 shows that the number of unit-tasks per episode ranged between one and four.

Table 7.5: Number of unit-tasks per episode

Unit-tasks per episode	Number of episodes
1	12
2	10
3	9
4	4

The initial demonstration by Ruth in Episode R1 illustrated the difference between human-computer interaction based on task related goals and more open ended discursive interaction. The top-level goal was to demonstrate the basic features of the use of *Bioview*.

Sub-goals were to demonstrate the execution of datacube operations and the use of the datasheet and graph_windows. However, the interaction did not correspond to well defined unit-tasks, and the episode consisted of a long unstructured sequence of operations, as illustrated by the following extract of the transcript of the discussion between Ruth and Tom:

> R: [*undo max_win(datasheet)*] And you can connect all this information into graphs here. [*con_graph(1) operation executed*]. Right, can change the type [*s_graph(b) operation executed*] and scale. [*sequence of scale_graph(sheet), scale_graph(row) and {scale_graph(cube)} operations executed*]
>
> T: Can you represent data on graph?
>
> R: Yeah, can also change other temperature by using, [*referring to T-value slider*] or you can use this arrow along here? [*referring to T-value slider, confirming datacube scale, and executing m_sheet(T) operation*]
>
> T: That's at 20 degrees.
>
> R: Yeah, that's going down again. [*referring to the decreasing T-value as m_sheet(T) operations are executed*]
>
> T: Right
>
> R: Change. [*sequence of arbitrary menu selections: {s_graph(b)}, scale_graph(sheet), scale_graph(row)*] Can also have pie charts like that. [*s_graph(p) operation executed followed by a {s_graph(p)} operation*] Can create new graphs like that [*con-graph window(2)*], if you knew that one had ben connected. [*referring to graph window(1)*] You could move it over here [*referring to the vacant bottom left region of the screen*] You can move things around, like that, [*resize_window(datasheet) operation executed*] so moves over, pressing Tidy tidies them up, [*"click" operation executed to bring the datasheet window to the top of the desk-top*] so can see everything [*Tidy menu item selected*]. Right.
>
> T: Can you actually plot this table here [*referring to the datacube*] on graph there. [*referring to graph window(2)*]?
>
> R: The cube?
>
> T: Yeah
>
> R: Well that's basically, if you [*sequence of s_sheet(L), confirm(scale-cube), and m_sheet(L) operations executed*] can change it around like that, so temperature is now along here [*referring to the T-columns on the datasheet*] [*con_graph(2) operation executed*]
>
> T: Right, so got temperature along the bottom now. [*referring to the horizontal axis of the rate/T graph displayed in graph window(2)*]
>
> R: Yeah can also move it about like this. [*m_row(C) operation executed*] Click light intensity up to make a line graph. [*s_sheet(L) and m_sheet(L) operations executed*]
>
> T: How did you do line graph again?

Without a structured task definition, the use of the software was not routine, and no planning was involved, making a hierarchical representation of the task and the associated cognitive procedures inappropriate.

With the exception of Episode 6, all the task related episodes in each of the sessions featured an attempt to apply one of the successful methods identified in Section 6.2.3. After correction for operations associated with manipulation errors and reduction to account for window manipulation and graph selection operations, the action string lengths (n_r) corresponding to the implementation of these methods were small (see Table 7.1). For example, the tri-modal values of n_r for Method 3, the most common method with 15 applications, were two, three and four operations within a range of two to six operations. In contrast n_r for the non-routine cognitive interaction in Episode 6 were typified by much longer action string lengths. Five of the seven action strings associated with this episode had corrected and reduced lengths greater than five, including lengths of 16, 17 and 46. Although it was possible to represent the non-routine interaction in Episode 6 in terms of a small number of unit tasks per episode, each unit-task was typically significantly more complex and less structured than the unit-tasks which were used to represent the routine interaction in the other episodes.

7.3.2 Identification of successful methods

The interaction in each of the observed sessions was structured by the set questions, making it relatively straightforward to divide each session into task related episodes. In general the identification of the method that was adopted in each episode was not problematic. However, Episode R4 illustrates that it may be possible to allocate more than one method to an episode.

In Episode R4 Ruth suggested to Tom that finding the optimum value of light or carbon dioxide was "basically the same as finding same level of temperature". This implied that she was recommending the use of the method she adopted in Episode R2, namely Method 1, and the goal structure of Episode R4 corresponded to this method. The sub-goal of Unit-task 1 was to maximise the C-value at the optimum temperature by executing an m_row(C) operation on a T-sheet located at the optimum temperature location:

> R: [...] If you click on temperature [*s-sheet(T) operation executed*] so got light intensity over here. [*referring to the horizontal axis of the rate/T graph displayed in graph window(2)*]
>
> T: I see, temperature's fixed now at 30. [*referring to the T-value display in graph window(2)*]
>
> R: Yeah.

T: If I try moving along here, shall I? [*m-row(C) operation executed*] Alright, go other way. [*m-row(C) operation executed*]

R:Yeah, need to go back.

T: Over here? [*scroll_win operation executed*]

R: [*m-row(C) operation executed to move the C-row to the maximum position*] Yeah. 0.1.

T: Is that the highest point we can go? [*referring to the maximum C-value for the displayed row*]

R: Yeah, that's maximum there. [*pointing to the rate value for the maximum L-value shown on the horizontal axis of the rate/L graph*]

T: Would seem to be here.

In Unit-task 2 the optimum temperature was changed to a maximum value by executing an m_sheet(T) operation.

R:Yes, haven't got maximum temperature so ...

T: If I go back here. [*referring to the T-value scroll box*]

R:Yeah

T: Click that [*redundant s_sheet(T) operation executed*] and increase temperature like that. [*m_sheet(T) operation executed*] Right, I think 40 is the maximum. So this one here ... [*referring to the rate value corresponding to the maximum L-value*]

The rate/L was inspected again in Unit-task 3 in order to determine the optimum value of the light intensity with the C-value and the L-value both fixed at a maximum:

R: Yeah so that is the ... [*indicating that the optimum value of the light intensity is now shown on the rate/L graph*]

Viewed in isolation as a complete episode in its own right Unit-task 1 could be interpreted as an example of the use of Method 3(T) - on the T-sheet corresponding to the optimum temperature execute m_row(C) operations and observe the effect on a rate/L graph to determine the optimum value of the carbon dioxide level. However, a consideration of the full sequence of three unit-tasks shows that the episode featured Method 1(T) - maximise the level of carbon dioxide by executing m_row(C) operations on a T-sheet, set the T-value associated with the T-sheet to the required value, and inspect the resulting rate/L graph to determine the optimum L-value.

This illustrates the importance of defining episode boundaries from a task perspective, as opposed to a system perspective. Even in the well-defined task context of the observed sessions, the allocation of an episode boundary was critical in the correct identification of the applied method. In more open, less structured contexts the allocation of episode boundaries would be even more critical.

7.3.3 Method selection

The significant display related method truncation in all three episodes indicates that display related selection rules were employed. Possible rules can be identified for each of the tasks that the learners were required to complete:

Determine the optimum value of the air temperature

a If the T-sheet is selected at the start of the episode select Method 2(T)
b If the L-sheet is selected at the start of the episode select Method 1(L)
c If the L-sheet is selected at the start of the episode select Method 3(L)
d If the C-sheet is selected at the start of the episode select Method 1(C)
e If the C-sheet is selected at the start of the episode select Method 3(C)

Determine the optimum value of the light intensity or explore the effect on the rate of photosynthesis of changing the light intensity

f If the L-sheet is selected at the start of the episode select Method 2(L)
g If the C-sheet is selected at the start of the episode select Method 1(C)
h If the C-sheet is selected at the start of the episode select Method 3(C)
i If the T-sheet is selected at the start of the episode select Method 1(T)
j If the T-sheet is selected at the start of the episode select Method 3(T)

Determine the optimum value of the carbon dioxide concentration or explore the effect on the rate of photosynthesis of changing the carbon dioxide concentration

k If the C-sheet is selected at the start of the episode select Method 2(C)
l If the L-sheet is selected at the start of the episode select Method 1(L)
m If the L-sheet is selected at the start of the episode select Method 3(L)
n If the T-sheet is selected at the start of the episode select Method 1(T)
o If the T-sheet is selected at the start of the episode select Method 3(T)

Compare the effect on the rate of photosynthesis of changing the light intensity and changing the carbon dioxide concentration

p If the L-sheet is selected at the start of the episode select Method 7(LL)
q If the C-sheet is selected at the start of the episode select Method 7(CC)
r If the T-sheet is selected at the start of the episode select Method 6(TT)

Table 7.6 shows both possible and observed use of selection rules for each application of a successful method. For each episode the applicable rules are marked as "•" and the observed use of a rule is marked as "o". An inspection of the table reveals that there were 29 occasions when it would have been appropriate to apply one of the rules. Method selection was consistent with the application of one of the rules on 18 occasions. This implies that the human-interaction was predominantly rule-based when an attempt was made to implement a successful method.

7.4 Summary

The analysis from a system perspective of the human-computer interaction observed during the laboratory sessions resulted in three findings. Firstly, the state of the system, as represented by the state of the display, was very influential in the selection of methods. As such the interaction history of a session was instrumental in determining future method selections. Secondly, the learners needed informative confirmation of the effects of their actions for a smooth flow of activity between the task and software domains. When the outcomes of direct manipulation were unexpected or unintelligible, goal directed cognitive behaviour was interrupted, compromising the efficient completion of the task in hand. Thirdly, learners misunderstood the relationship between the current state of the system and the representation of this state by the display. In particular there was a widespread misconception of the relationship between direct manipulation of the display and changes of the active system sub-register. This finding contrasts with the superficial impression of a close match between learner and designer models implied by the learners' extensive selection of successful methods.

The analysis from a task perspective provided little evidence that the *interaction* between variables was considered by the learners. The relationship between variables was interpreted in a simplistic fashion, as indicated by the extensive task related method truncation in each session. In particular it appeared the learners did not possess, or were incapable of applying, an understanding of the concept of limiting factors in photosynthesis. There was some evidence of preferential application of direct manipulation techniques. In tasks in which a specific value of a variable needed to be determined, direct manipulation tended to be used to fix the values of one or both of the other variable, with the specific value determined by the conventional inspection of a graph. When the task involved the exploration of the effects of varying a variable, direct manipulation tended to be used to vary the value of the variable.

Table 7.6: Use of display related selection rules

Episode	Selection Rule a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r
D2	•	•	•	•	•													
D3A						•	•	•	•	•								
D3B											•	•	•	•	o			
D4A						•	•	•	o	•								
D4B											•	•	•	•	o			
D5																•	•	o
D6																-	-	-
D7	•	•	•	•	•													
D8	•	•	•	•	•													
D9A						•	•	•	•	o								
D9B											•	•	•	•	o			
D10						•	•	•	•	o								
D11																•	•	o
D12																-	-	-
Tot (•)	3	3	3	3	3	4	4	4	4	4	3	3	3	3	3	2	2	2
Tot (o)	0	0	0	0	0	0	0	0	1	2	0	0	0	0	3	0	0	2
S2	•	o	•	•	•													
S3A											•	•	•	•	•			
S3B						•	•	•	•	o								
S4						•	•	•	•	o								
S5																•	•	o
S6																•	•	•
S7	o	•	•	•	•													
S8A											•	•	•	•	o			
S8B						•	•	•	•	o								
S9A											•	•	•	o	•			
S9B						•	•	•	•	o								
S10																-	-	-
S11																•	•	•
Tot (•)	2	2	2	2	2	4	4	4	4	4	3	3	3	3	3	3	3	3
Tot (o)	1	1	0	0	0	0	0	0	0	4	0	0	0	1	1	0	0	1
R2	•	o	•	•	•													
R3A											•	•	•	•	•			
R3B						•	•	•	•	•								
R4						•	•	•	•	•								
R5																-	-	-
R6D																•	•	•
Tot (•)	1	1	1	1	1	2	2	2	2	2	1	1	1	1	1	1	1	1
Tot (o)	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Tot (•)	6	6	6	6	10	10	10	10	10	10	7	7	7	7	7	6	6	6
Tot (o)	1	2	0	0	0	0	0	0	1	6	0	0	0	1	4	0	0	3

The application of the GOMS approach to describe the human-computer interaction in the laboratory sessions was effective as a method for describing the interaction. It was possible to regard each session as a sequence of goal related episodes. With the exception of the initial illustrative episodes, each episode consisted of a limited number of well defined unit-tasks. These unit-tasks consisted of a limited number of operations when successful methods were applied. This was not the case for the application of other idiosyncratic methods; these consisted of long convoluted action strings. The selection of methods appeared to be predominantly rule based.

Chapter 8

Learner and designer models in the use of direct manipulation educational software

In this chapter an application of the Jigsaw Model is used as a basis for discussing learner and designer models in the use of direct manipulation educational software. In Section 8.1 the use of the model to provide a structure for evaluating the cognition associated with the use of *Bioview* to explore limiting factors in photosynthesis is discussed. The application of the model to evaluate the use of *Bioview* observed in the laboratory sessions is used in Section 8.2 to critique the design of *Bioview* and suggest design improvements. In Section 8.3 the results of applying the model are used to discuss learner and designer models in the use of direct manipulation educational software.

8.1 Using the Jigsaw Model to evaluate the use of *Bioview* with the PSYNTH datacube.

The use of the Jigsaw Model is intended to provide both an indication of design problems and an explanation of their origin. The use of the explanatory component of the model, the jigsaw framework, is discussed in Section 8.1.1, with reference to each of the seven design issues associated with the framework. The predictive component, the GOMS approach, is discussed in Section 8.1.2. Figure 8.1 illustrates the use of the model with *Bioview* and the PSYNTH datacube.

8.1.1 Application of the jigsaw framework for cognitive complexity

The jigsaw framework is based on the adoption of an "inner" psychological theory and a well defined representation of the system. Research into students' understanding of photosynthesis (see Section 3.1) has typically adopted a constructivist approach, that is, students are regarded as active and purposeful individuals who learn by a process of constructing and reconstructing concepts. This indicates a constructivist view of cognition as an appropriate "inner" theory. Adopting this approach implies a learner's model based on misconceptions of how environmental factors affect photosynthesis, and a designer's model of the learner's model based on learners correcting their misconceptions through the reconstruction of their concepts.

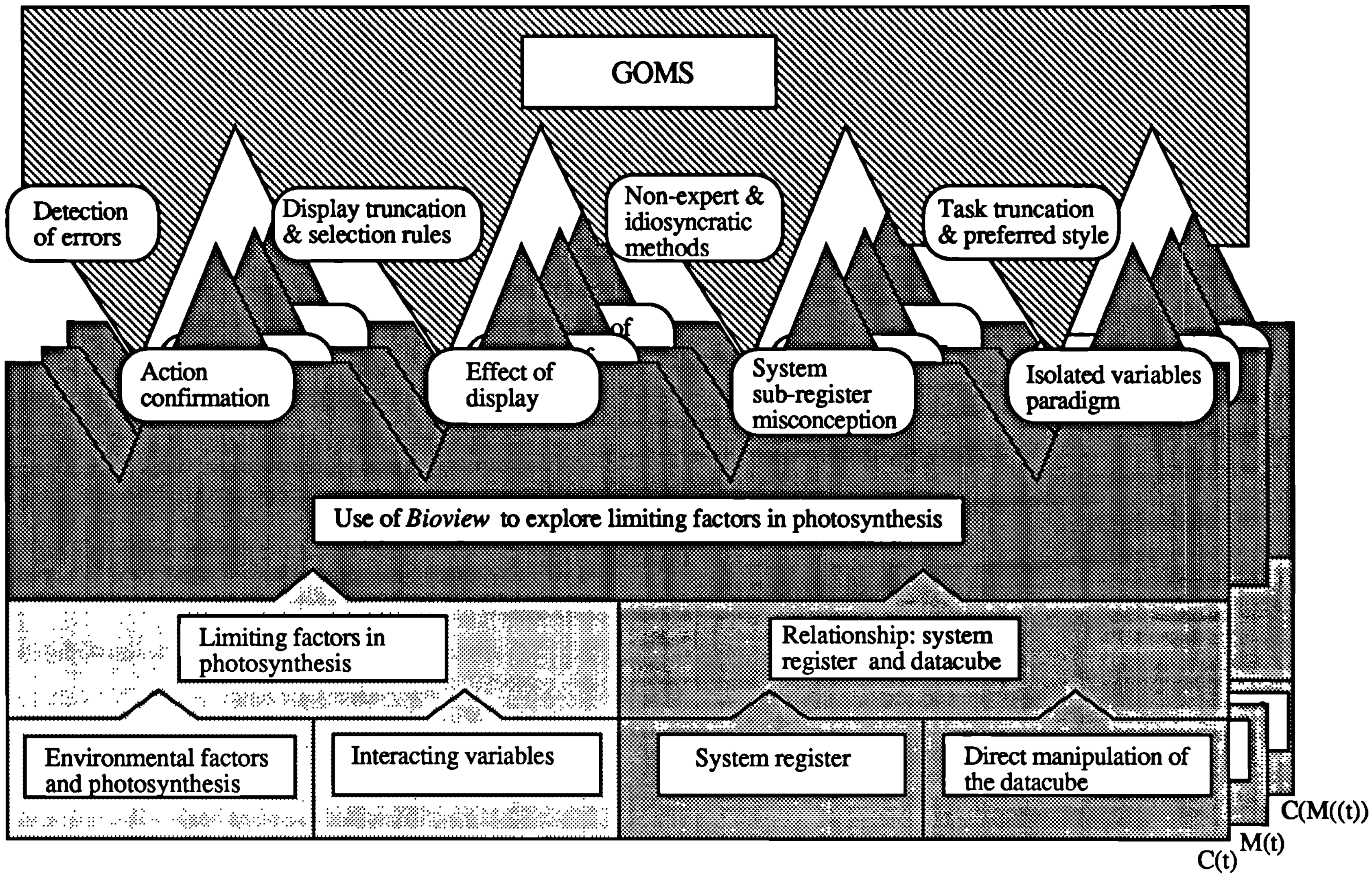


Figure 8.1: Application of the Jigsaw Model to the use of Bioview to explore limiting factors in photosynthesis

The function of the system is represented metaphorically by the datacube, and a structural model is provided by the lexically scoped system register. This implies that the designer's model of the learner's model assumes that learners will perceive the functionality of *Bioview* in terms of the direct manipulation of the datacube and that they will instinctively assume lexical scoping of the datasheet values.

In the following sections, each of the seven design issues in the Jigsaw Model are discussed in terms of the designer's model, the learner's model, and the designer's model of the learner's model. The results of these discussions are summarised in Table 8.1.

8.1.1.1 Concepts of how environmental factors affect photosynthesis

The designer's model of how photosynthesis is affected by environmental factors is based on an appreciation that light, carbon dioxide, and temperature are critical factors in photosynthesis; an appreciation gained through an understanding of the role that each of these factors plays in the process of photosynthesis.

The learner's model will typically incorporate misconceptions of the role of light and carbon dioxide and the significance of temperature in photosynthesis. Misconceptions about the role of light and carbon dioxide are extensively reported in the research literature, but there is no reported research on students understanding of the significance of atmospheric temperature. The studies conducted by Roth et al. (1983), Wandersee (1983), Bell & Brook (1984), Smith & Anderson (1984), Stavy et al. (1987), and Eisen & Stavy (1988) indicate that students typically only have a superficial awareness of the role of light; they are aware that plants need light to grow and be healthy, but they are unaware that light is essential in the production of plant food. Many students do not know that atmospheric carbon dioxide is the source of carbon which is reduced during photosynthesis to produce sugar. This leads to widespread misconceptions as to what constitutes a plant's food (Simpson & Arnold, 1982a, 1982b; Roth et al., 1983; Soyibo, 1983; Wandersee, 1983; Bell & Brook, 1984; Driver et al., 1984; Smith & Anderson, 1984; Eisen & Stavy, 1987, Stavy et al., 1988; Barker & Carr, 1989). Students are also confused about the absorption of carbon dioxide and the production of oxygen (Wandersee, 1983; Soyibo, 1983; Stavy et al., 1986, Eisen & Stavy, 1987). For example, Soyibo (1983) in an investigation into the misconceptions held by students aged 15-19 years reports that 60% of the students in the sample thought that oxygen passes into plant leaves and carbon dioxide passes out.

The designer's model of the learner's model needs to recognise that these common misconceptions may be influential in learners' understanding of how environmental factors affect photosynthesis, and appropriate ways of reflecting this recognition in the design of software, and associated documentation, should be considered.

Table 8.1: Jigsaw framework applied to the use of Bioview with the PSYNTH datacube

Level	Issue	Designer's model	Learner's model	Designer's model of the learner's model
1	1	Environmental factors affect photosynthesis.	Misconceptions about how environmental factors affect photosynthesis.	Misconceptions about how environmental factors affect photosynthesis; personal construction of concepts.
1	2	The idea of interacting variables; need for multi-variable exploration.	No concept of interacting variables; isolation of variables paradigm.	Typically no appreciation of interaction between variables and use of an isolation of variables paradigm.
1	3	System register for each datasheet.	System register for the datacube.	Instinctive appreciation of the datacube in terms of datasheets.
1	4	Functional models based on direct manipulation of the datacube and dynamic data exchange between datacube, datasheet and graph windows.	Functional models based on direct manipulation of the datacube and dynamic data exchange between datacube, datasheet and graph windows.	Learners understand how to directly manipulate the datacube and the linking to graph windows.
2	5	Factors which affect the rate of photosynthesis interact with each other to limit the rate.	Misconceptions about limiting factors in photosynthesis.	Learners have misconceptions about limiting factors in photosynthesis.
2	6	Active system sub-register (lexical scoping).	No concept of an active system sub-register (dynamic scoping).	Learners will implicitly understand effects of datacube operations on the system state in terms of lexical scoping.
3	7	Using *Bioview* with PSYNTH datacube represents how the rate of photosynthesis changes with environmental factors.	No correction of misconceptions ; use of isolation of variables paradigm.	Using *Bioview* with PSYNTH datacube corrects misconceptions and encourages multi-variable exploration.

8.1.1.2 General concepts concerned with interacting variables

The designer's model assumes that variables are typically correlated and that behaviour is very often determined by the interaction between variables. However, as Lucas & Tobin (1987) and Dawson & Rowell (1986) point out, students typically adopt an approach which can be described as "change one thing at a time and keep all else constant" or "isolation of variables". As such the designer's model and the learner's model of the general task domain are often fundamentally different. The designer's model of the learner's model must take account of this difference, as illustrated by the science curriculum materials concerned with variables developed by Adey, Shayer & Yates (1989).

8.1.1.3 *Bioview* system representation

The concept of a system register provides the basis for the designer's surrogate structural model of the system. As explained in Chapter 5 this register consists of three separate sub-registers; one for each sheet in the datacube. Each sub-register holds the current value for each variable for the appropriate datasheet. The human-computer interaction during the laboratory sessions indicated that the learners held a simpler surrogate model of a single register for the datacube. In terms of this model the state of the system is represented by a value for each of the variables, for example a C-value, a L-value, and a T-value, rather than the total of nine values related to the three system sub-registers. The lack of any discussion of a surrogate model in the documentation accompanying *Bioview*, and the failure of the interface to provide an explicit representation of a surrogate model, both indicate that the designer's model of the learner's model assumes an implicit appreciation of the system structure and its state.

8.1.1.4 *Bioview* functionality

As the direct manipulation paradigm provides the interface design rationale for *Bioview* the functional aspects of the designer's model will be based on direct manipulation techniques. In addition, the designer's model assumes that the graphical representation of the three interacting variables is informative and intelligible. The ease with which the datacube was manipulated during the laboratory sessions indicates that the direct manipulation paradigm and the visual affordance provided by the datacube were acceptable in terms of a learner model, implying some correspondence between the functional aspects of designer and learner models. The designer's model of the learner's model is that direct manipulation of visually presented objects provides an effective and understandable way of interacting with computers in a general sense, and that specifically the datacube operators provide a functionality tuned to the task of exploring the relationship between interacting variables.

8.1.1.5 Limiting factors in photosynthesis

The designer's model will take account of how environmental factors interact to establish the optimum conditions for photosynthesis. In particular this model will recognise how a factor limits the rate of photosynthesis until its value is increased to a non-limiting saturation value, with the possibly of a further increase in the rate resulting from a change in the value of one of the other environmental factors. The research by Amir & Tamir (1989, 1990) indicated that the learner's model will often be based on misconceptions, notably the belief that a limiting factor limits the rate of photosynthesis after the saturation value has been reached rather than before it is reached. The designer's model of the learner's model will need to take these misconceptions into account. In the case of *Bioview* this is attempted through the the worksheet questions designed to guide the use of the PSYNTH datacube.

As noted in Chapter 7 there was very little evidence that the learners observed during the laboratory sessions used the concept of limiting factors. This is particularly illustrated by considering their approaches to Question 5. This question could have been answered by using the concept of limiting factors. In the PSYNTH datacube the maximum value of the light intensity is very close to a non-limiting value, but the level of carbon dioxide is still clearly limiting the rate of photosynthesis, making it best to increase the level of carbon dioxide to achieve a further increase in the rate of photosynthesis. Rather than use this approach, the learners simply compared the gradients of rate/L and rate/C graphs. This behaviour is very similar to that observed by Amir & Tamir (1989) in response to a similar question they posed, that is whether species A (close to a non-limited state for the given maximum light intensity) or B (clearly limited at the same maximum light intensity) would be best to grow in shaded conditions. Amir & Tamir reported that only 3% of their research sample applied the concept of limiting factors to this question.

8.1.1.6 Relationship between the datacube display and the system register

The values corresponding to each datasheet are lexically scoped, that is a change in the values in one sub-register does not result in a change in the other sub-registers. This implies that the designer's surrogate model of the system uses the concept of an active system sub-register, that is, the system sub-register corresponding to the currently selected datasheet. Thus the datacube is seen by the designer as representing three different states of the system, with each of these states corresponding to a different datasheet. The perception of the system in terms of a single three value register model, which represents the state of the datacube, implies that learners' will assume dynamic scoping of the datacube variables, so that a change in the state of any datasheet is automatically reflected in the change in the state of the other datasheets. The designer's model of the learner's model assumes that the user will perceive the datacube as a representation of three different states of the environment, as opposed to a representation of one state of the environment which would have resulted by yoking the system and task space in terms of dynamic scoping.

8.1.1.7 Use of *Bioview* to explore interacting variables in photosynthesis

The designer's model assumes that direct manipulation of the datacube is an effective way of encouraging students to change more than one variable at the same time in order to explore the relationship between three interacting variables. In fact, using the datacube in this way involves direct manipulation of one variable by executing m_sheet or m_row/col operations, and the "indirect" manipulation of a variable by inspecting a rate/variable graph. The laboratory observations of the use of *Bioview* by the teacher-students and the students indicated that they persisted in adopting an isolation of variables strategy. It would seem that the functional models held by the teacher-students and the students during the laboratory sessions were at variance with the designer's models in this sense. However, the designer's model of the learner's model assumes that a mix of direct and indirect manipulations is effective in encouraging a multi-variable approach. It also assumes that graphical, as opposed to tabular, representation of the variation of rate of photosynthesis is preferable. This second assumption is consistent with the finding of Amir & Tamir (1989) that students displayed a greater understanding of limiting factors in photosynthesis when data were presented in a graphical form rather than a tabular form.

8.1.2 Application of the GOMS model

The application of the GOMS model is intended to provide the indicative component of the Jigsaw Model. The documented adaptations of successful extensions of GOMS (Kieras & Polson, 1985; John, 1990; Gray, John & Atwood, 1993) were significant factors in the choice of the GOMS approach as a indicative component. However, as noted by Olson & Olson (1990) there are serious questions about the value of the GOMS approach, which they summarised in the form of a compilation of the shortcomimgs originally identified by Card et al. (1980b):

1 The model applied to skilled users, not to beginners or intermediates. Such *nonskilled* users spend considerable time engaged in problem-solving activities, rather than simply retrieving and executing plans, and move smoothly between problem solving and skilled behaviour.
2 The model gave an account of skilled performance as an asymptote but no account of either *learning* of the system or its recall after a period of disuse, nor how to design an easily used consistent interface.
3 The model focused on errorless performance and, thus, gave no account of the *errors* that frequently occur even in skilled performance.
4 The model was most explicit about elementary perceptual and motor components of skilled behaviour but tended to treat the *cognitive processes* in skilled behaviour in a less differentiated fashion.
5 The model was developed exclusively for tasks in which the principal components that were being modelled could reasonably be assumed to be serial in nature. However, tasks, have a substantial number of component processes that, at some level, must occur in *parallel*..
6 The model does not address *mental workload* - how much must be held in mind while using the system.
7 The model addresses only the usability of a task on a system and does not address *functionality*, that is, what tasks should be performed by the computer.
8 The model does not address the amount and kind of *fatigue* users experience using the system.
9 The model does not account for individual *differences* between users.
10 The model does not provide guidance in predicting whether users will judge the system to be either useful or satisfying, or whether the system will be globally *acceptable*.
11 The model stops short of addressing any aspects of how computer-supported work fits or misfits office or organisational life.
(Olson & Olson, 1990, p.227)

In addition to these shortcomimgs there is a common feeling that the GOMS approach is inappropriate for the representation of interaction in display based environments. Howes & Payne (1990) typify these feelings. In support of their criticism of the application of the GOMS model to represent display based interactions they quote the finding of Mayes, Draper, McGregor & Oatley (1988) that experienced users of *MacDraw* could not recall the names of menu items that they regularly used. They argue that the finding of Mayes et al. demonstrates that users rely on the screen presentation and do not need to remember the menu names. This leads them to claim:

> A GOMS model does not need to check the wording of the menu items before enacting a menu selection: the sequence of actions to achieve the current goal is already explicitly determined. Such a model must predict that skill is associated with complete knowledge of the steps in a method, and so cannot cope with the results of Mayes et al. (1988). (Howes & Payne, 1990, p. 638)

Some of these shortcomimgs have been attended to by the extensions to the basic GOMS model. For example, the number of production rules and level to which they are nested in a CCT model provides an estimate of mental workload (Kieras & Polson, 1985), thus addressing the sixth shortcoming, and CPM-GOMS (John, 1990) attempts to model parallel processing, thus addressing the fifth shortcoming.

The Jigsaw Model attempts to provide more comprehensive attention to these shortcomings from an educational perspective. The limited treatment of cognitive processes (fourth shortcoming) is a serious omission in an educational context. This is dealt with by the inclusion of an inner psychological theory in the jigsaw framework for cognitive complexity. The restriction to expert performance (first shortcoming) and errorless performance (third shortcoming), the lack of attention to functionality (seventh shortcoming), and the failure to attend to individual differences are also significant omissions in an educational context. As explained in the following sections the representation of a GOMS analysis in terms of an action string makes it possible to cope with these shortcomings, thus enabling an identification of design problems for explanation in terms of the jigsaw framework. In addition this representation assists in a consideration of the effect of the screen display.

8.1.2.1 Non-expert behaviour

In any context, it is likely that different users of a software application will have different levels of expertise and experience. Black et al. (1987) have identified levels of expertise in the context of text editing, which can be used to provide a general scheme for considering the expertise of a user. At the first level, users rely on preconceptions formed from prior experience; a finding reported in the first instance by Sebrechts, Black, Galambos, Wagner, Deck, & Wikler (1983). In terms of the GOMS approach the function of operators at this level is simply perceived in terms of previous experience. The second level is characterised by initial learning with a simple link assumed between goals and operators. This is illustrated in terms of the GOMS model by the finding of Kay & Black (1984, 1986) that goals and operators are acquired first. Black et al. summarise these findings with respect to text editing as follows:

> Novices seem to conceptualise the commands merely by what goals they are relevant for accomplishing. Because at this level of specificity they have not yet acquired the procedures or plans that are associated with text editing, each text editing task becomes a problem solving task in which they must actively search through their representations of the commands and construct the sequence of commands necessary to accomplish the task. (Black, Kay & Soloway, 1987, p. 47)

At the third level, plans are formed which correspond to GOMS methods. The fourth level is characterised by the combination of plans and the use of selection rules.

The use of educational software by learners with the first two levels of expertise must necessarily be non-expert. However, the use of software by learners who can form plans and apply methods (levels three and four) may also be non-expert; in particular potentially successful methods may be inexpertly applied or inappropriate selections may be made. An indication of the non-expert application of potentially successful methods can be achieved by comparing the action strings for the observed execution of a method with the corresponding action string for an expert application of the method (see Section 7.1.3 for an example of the application of this technique). In order to facilitate this comparison the observed action string should be corrected for errors and reduced to delete operators simply concerned with configuring the appearance of the screen. As expert performance is representative of the designer's model, and non-expert performance is representative of the learner's model, a consideration of the expert and non-expert action strings allows a comparison between learner and designer models.

The action string representation allows the GOMS model to be extended to consider non-expert performance; but this extension is limited to a consideration of relatively experienced users, that is users at expertise levels of three and above. In this sense the use of GOMS is still restricted in an educational context, as users in this context are often novices or initial users. However, task related learning sponsored by software use is unlikely to occur until computer intrinsic skills are acquired. This will typically coincide with level three or above expertise, making comparison of expert and non-expert strings relevant to the evaluation of educational software.

8.1.2.2 Commission of errors

Errors are a common feature of both expert and inexpert use of software. For instance, Card et al. (1983) reported that 26% of the time spent in the expert performance of text editing was spent in correcting errors. Usually, it is possible to imagine errors originating from a number of sources. Wright, Monk & Cary (1991) provide a comprehensive categorisation of error sources as follows: (i) easily rectified slips, which can be attributed to a failure in attention; (ii) misleading cues provided by the software; (iii) attempts to find an action which is not provided by the software, (iv) hidden functionality when the cues provided by the software are insufficient to guide the user, (v) limiting or inappropriate functionality when users have to compromise their actions; and (vi) missing or ambiguous feedback. Slips are of no great significance and so corresponding errors can simply be omitted from the action string without further consideration. Attempts to find non-existent functionality or to use limited functionality are a matter of fundamental design and the errors associated with these sources would manifest themselves as confused method applications, rather than display or dialogue related errors. The remaining three types of sources of error (misleading cues, hidden functionality, and missing or ambiguous feedback) can be used to categorise the error related actions deleted from the action string, thus providing a basis for further analysis.

The representation of human-computer interaction by an action string enables display and dialogue operations which are committed in error to be easily identified, allowing the error free interaction to be represented by the action string corrected for the observed errors. The corrected action string can then be used as a basis for identifying the application of well defined methods. In this sense the process of correcting the action string for display and dialogue errors makes it possible to apply the GOMS approach. However, the identification of the errors in itself provides valuable information on the character of the human computer interaction represented by the action string, and by categorising the nature of the errors it may be possible to identify weaknesses in the design of the software. For example, most of the display and dialogue errors observed during the use of *Bioview* were related to the need for action confirmation, and as shown in Section 8.2.2 a consideration of this phenomenon can be used as a basis for suggesting design changes.

Thus the combination of error corrected action strings and the error categorisation suggested by Wright et al. provides the possibility of extending the GOMS approach to deal with interactions which contain errors.

8.1.2.3 Functionality

The functionality of a software application should be appropriate to the task in hand, that is the processes that the software supports should be relevant and meaningful to the nature of a chosen task. Software functionality is evident to the user in terms of the operations that it is possible to execute. In this sense, these operations are not neutral; their style and character may significantly influence the way in which the software is used. This is illustrated by the argument in Section 8.2.4 that if direct manipulation is a preferred style of interaction, the fact that only one variable can be varied by direct manipulation in Methods 2 and 3 may encourage learners to adopt an isolation of variables paradigm. Thus the preferred use of operators can be used to indicate the learner's perception of software functionality.

An indication of a user's preferred style of use is given by the extent and character of task related method truncation, that is the reduction of a method due to a user's perception of the task in hand. This is illustrated in Section 7.2 by the task truncation by learners of Methods 2 and 3 due to the omission of the inspect_graph operations in these methods. Thus the GOMS approach can be extended to attend to the functionality shortcomimg by inspecting action strings of observed human-computer interaction for evidence of task related truncation.

8.1.2.4 Individual differences

People learn in different ways. Indeed this is a fundamental assumption of constructivism, the "inner" theory adopted in this application of the Jigsaw Model. As such it is important to be able to accommodate individual differences. To some extent the comparison of expert and non-expert applications of successful methods allows for individual differences. However, some methods adopted by learners will be idiosyncratic, without any clear resemblance to an established successful method. This obviously makes a direct comparison of learner and designer models in terms of non-expert and expert application of successful methods impossible. However, an analysis of the action string corresponding to an idiosyncratic method can be used to illuminate the learner's model, and thus enable an indirect comparison of learner and designer models. This is illustrated in Section 7.1.4 in which the analysis of the actions strings corresponding to the long and confused attempt by Ruth and Tom to answer Question 5 yielded convincing confirmation of the existence of the active system sub-register misconception.

8.1.2.5 Significance of the display

It is commonly acknowledged that direct manipulation interfaces should be based on the manipulation of visually presented objects (for example, see Shneiderman, 1987). Given that the style of the operators afforded through a direct manipulation interface may be influential in the formation of the learner's model of a system's functionality, it is likely that the form of the display will be similarly influential in the formation of this model. Thus it is critical that the GOMS model is extended so that it can in some respects attend to the significance of the display.

In fact, the selection element of the basic GOMS model does provide a limited way of considering the effect of the display, as illustrated in Section 7.3.3 in which selection rules for the use of the PSYNTH datacube with *Bioview* were proposed in terms of the state of the display. However, the notion of display related method truncation allows the effect of the display to be considered in a novel fashion. When display-truncation occurs, the user has decided to adopt a particular method because it is implied by the existing state of the display. For example, if a T-sheet is selected and the learner wishes to explore the effect of changing the temperature, Method 2 is implied. Hence the extent of display-truncation indicates the overall significance of the display in method selection, and the form of the display truncation indicates the specifics of the way the display affects method selection.

Thus the basic GOMS structure can be augmented to represent the influence of the display by inspecting action strings corresponding to observed methods to ascertain the extent and character of display related truncation.

8.2 Results of applying the Jigsaw Model

The analysis in Chapter 7 resulted in the following findings: (i) the state of the display had a marked influence on the selection of methods by the user, (ii) the disruption by the need for action confirmation induced errors and a disruption of smooth activity flow during the interaction cycle, (iii) the existence of a mismatch between the device models employed by the designer and the users, and (iv) preferred uses of direct manipulation. These issues are discussed in this section.

8.2.1 Effect of the display

The extensive display-truncation and existence of display based selection rules indicates that the state of the system was very influential in method selection. This influence is paradoxical in that it can have both good and bad effects. From a positive point of view a well designed display can point the way to the execution of the next appropriate operation or operations. On the other hand, the influence of the display could constrain the selection of methods by the learner, resulting in a limited perception and use of the functionality of the system.

The "display paradox" can be illustrated with reference to *Bioview*. For example, Methods 1, 2 and 3 can all be used to determine the optimum value of the temperature. Methods 2 and 3 both rely on detecting the optimum temperature by observing a sequence of rate/L or rate/C graphs and identifying the instance of a graph which represents the optimum conditions for photosynthesis. In Methods 3(C) and 3(L) it is only possible to determine the value of the temperature in terms of the discrete values associated with respectively a row or column. Similarly in Method 2(T) the values of the temperature are expressed in terms of the discrete values associated with the T-sheets. Of course, the optimum temperature may correspond to a value between the values associated with two adjacent sheets, rows or columns. Thus there is an inherent limitation in the precision with which the learner can identify the optimum temperature when Method 2 or 3 is used. However, in Method 1 the optimum temperature is determined by inspecting a static rate/T graph which shows the rate of photosynthesis plotted as a continuous variable against the temperature. In this case, it is possible to precisely identify the optimum temperature by inspecting the rate/T graph. If the T-sheet is highlighted at the start of an episode concerned with identifying the optimum temperature, it is likely that Method 2 will be chosen. When a precise identification of the optimum temperature is not required the influence of the display can be seen as positive; selecting Method 2 with the datasheet chosen by default makes the application of the method straightforward and efficient. However, if the requirement is for a precise identification of the optimum temperature, the influence can be seen as negative in encouraging the use of an imprecise method.

8.2.2 Activity flow

The display of null graphs clearly disrupted activity flow. Both teacher-students and students were confused by the appearance of these graphs, and resorted to arbitrary execution of operations in an attempt to "make" a graph appear. Carroll, Kellogg & Rosson (1991) observed a similar need for action confirmation in their study of the use of the *Training Wheels* instructional package for the *Displaywriter* word processing package. The need for action confirmation manifested itself as a "Did that work?" concern cycle which "often led users to redundantly specify option changes" (Carroll et al, 1991, p. 97). The execution of redundant operations in response to the appearance of a null graph distracted attention from achieving the current sub-goal; typically fixing the value of a variable or inspecting changes in a rate/variable graph as the value of a variable was changed. This led to the application of the currently selected successful method being compromised.

Action confirmation is in some senses similar to the idea of a "production paradox" introduced by Carroll & Rosson (1987). Adopting the attack strategy described in Section 2.1 would involve a complete re-design of *Bioview* along lines which are probably not compatible with the anticipated use of *Bioview*. However, it is reasonably easy to see how mitigation could be used to soften the effects of action confirmation. For example, rows or columns corresponding to sets of zero rate values could be hidden from the user; they would simply not be shown in the datasheet display. Another approach could be to block the use of a row or column corresponding to a set of zero values. If a user tried to highlight one of these rows or columns, an explanatory message would appear to state that access was denied and why it was denied. Both of these approaches are open to criticism; hiding some of the data from the user destroys the integrity of the database and presents an inaccurate representation of the relationship between the interacting variables, and the explanatory messages associated with the blocking approach would themselves compromise smooth activity flow. A better approach to mitigation would probably be to dim rows and columns in a similar fashion to the way that non-active menu items are dimmed in the Macintosh desktop user interface. This would signal to users that the selection of these rows and columns would be inadvisable.

8.2.3 User and device models

The extensive application of successful models implies a match between learner and designer goals. As it can be assumed that the designer holds an accurate system model, this indicates a match between learner and designer models. However, the existence of an "active system sub-register misconception" implies that there was a mismatch between learner and designer system models, indicating that any match was only in functional terms. This is illustrated by representing both models by "goal trees" as in Figure 8.2.

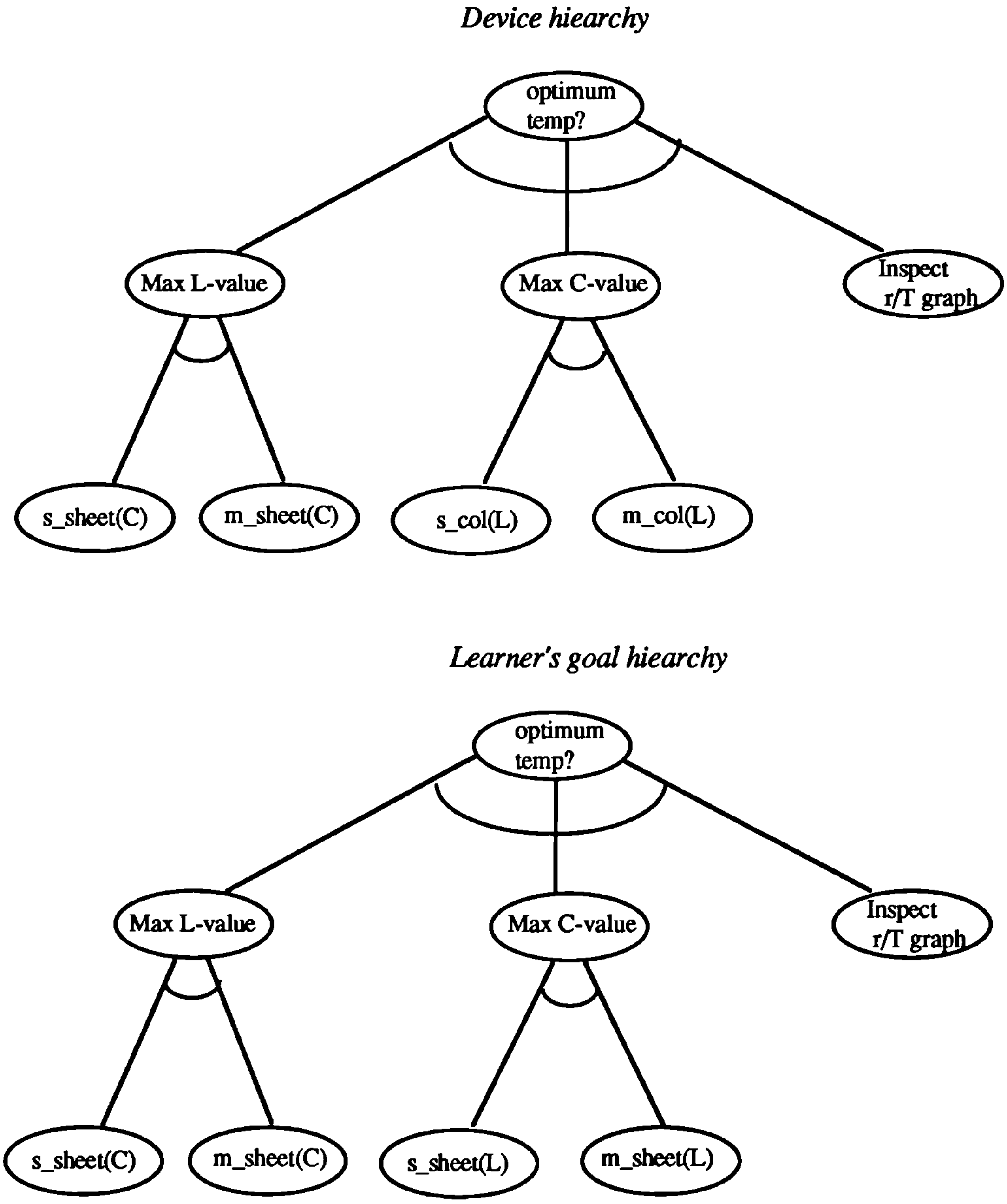


Figure 8.2: Learner and device goal trees for a misconceived application of Method 1(C)

This method was first used by Kieras & Polson (1985) in their study of user complexity in text editing systems, and later used by Sutcliffe & Springett (1992) and Springett & Grant (1993) to consider *MacDraw* users' models.

In Figure 8.2 an application of Method 1(C) is represented when the learner holds the active system sub-register misconception. The identical pattern of the goal trees for the device and the learner illustrates the match between the designer's and learner's goals. However, the decision by the user to maximise the value of the light intensity by manipulating the position of the L-sheet rather than moving the position of the L-column on the C-sheet, corresponds to the active system sub-register misconception, leading to problems in achieving the final sub-goal of inspecting a rate/T graph.

The active system sub-register misconception indicates that in terms of the YSS model (Payne, 1987) the task and device models are not yoked, ensuring that there is a semantic distance between the learner's perception of the task and the system's representation of the learner's actions. A more effective yoking between these models could be achieved by either modifying the goal structure of the learner or by changing the software. As the goal structure of the user is correct, in that it corresponds to the designer's goal structure, the first approach is obviously inappropriate. Thus a revision of *Bioview* to cope with the active system sub-register misconception must involve a redesign of the software. This revision could be attempted in two ways; either a surrogate model of the device structure could be explicitly represented to the user in anticipation that this would prevent the active system sub-register from occurring, or the way in which active direct manipulation operations affect the values in the system sub-register could be changed.

The first approach relates to how *Bioview* makes use of mitigation to cope with the assimilation paradox. The design is based on the assumption that users will find their prior conceptions of the geometrical structure of a cuboid helpful in understanding the relationships between three interacting variables. This approach, however, resulted in the active system sub-register misconception. Perhaps an attack strategy based on presenting a full and accurate model of the system should be adopted. This could simply consist of the inclusion of a display register window in which the variable values of each sub-register are displayed. As the datacube is manipulated these values would change.

The second approach can be understood in terms of the way that variables are scoped in *Bioview*. The effects of the active datacube operators (m_sheet, m_row/col) on the system state (as defined by the system register) are lexically scoped, that is, the change which results from executing an active operation is limited to the environment in which the operation is executed. This means that the changes are only relevant to the currently selected sheet, and thus the currently active system sub-register. If the active operators were dynamically scoped, a change resulting from the execution of an operator would be relevant to all three sheets, and result in a change in all three system sub-registers.

diSessa (1985, 1991), in a discussion of the scoping of *Boxer*, claims that dynamic scoping is more suitable for structural, as opposed to functional, models. The considerable problems that were observed in comprehending the system register structural model support diSessa's proposal. If *Bioview* was dynamically scoped, the concept of an *active* system sub-register would not exist, preventing the occurrence of the active system sub-register misconception. There are other implications associated with dynamic scoping. It can be argued that dynamic scoping would be a better representation of interacting variables, that is, the basis for a better functional model. With lexical scoping each orthogonal view represents a different state of the system in an unconnected fashion without stressing the interaction between the variables. With dynamic scoping each view would represent the same state of the system, stressing the interaction between variables.

8.2.4 Preferred style of interaction

Analysis of the data from a task perspective indicates, that the teacher-students and the students were unaware of the interaction between the variables, leading to an adoption of the isolation of variables strategy described by Lucas & Tobin (1987) and Dawson & Rowell (1986). The preference that users had for using direct manipulation techniques to vary a variable indicates that the style and extent of the available direct manipulation techniques may have been influential in the development of learners' goal structures of the task in hand. In this sense the style of direct manipulation afforded by *Bioview* may have encouraged an isolation of variables approach. As it is only possible to directly manipulate the datacube to change the value of one variable at a time, with changes in a second variable indirectly "manipulated" by inspecting a rate/variable graph (as in Method 2 and Method 3), users may have been encouraged to think of this as the most appropriate way to explore the effect of changing the variables; perhaps reinforcing their inherent hierarchical goal based perception of the task. The representation of the interaction between two variables is left to users; either they have to interpret the interaction in terms of the animated sequence of rate/variable graphs, or they have to compare instances of graphs shown in multiple graph_windows. This analysis suggests two ways in which the design of *Bioview* could be changed to support a consideration of the interaction between variables. Firstly, the presentation of the results of manipulating the datacube could be improved. Secondly, the direct manipulation operators could be changed so as to encourage the option of simultaneously directly manipulating two or three variables.

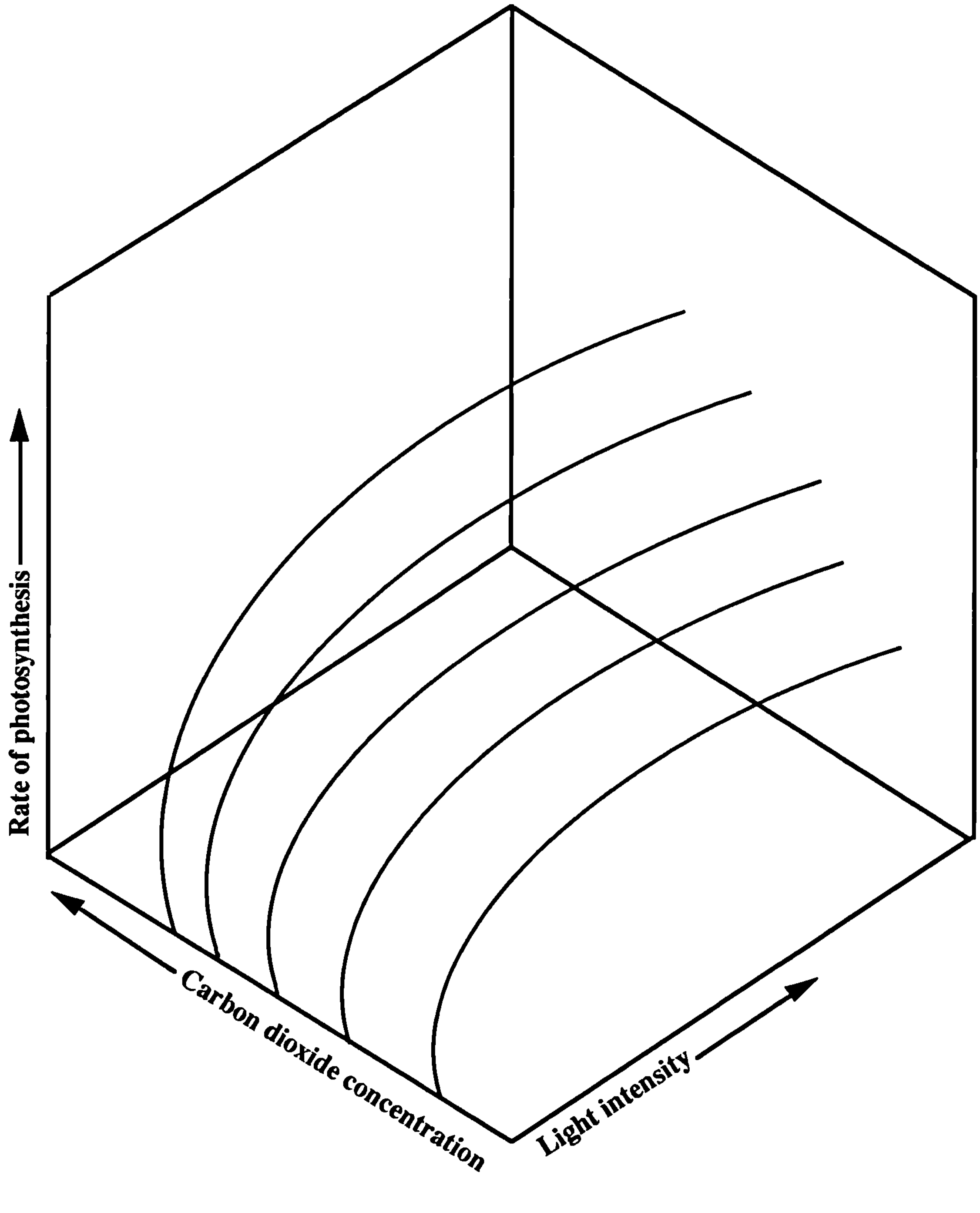


Figure 8.3: Using a three dimensional representation of the effects of the direct manipulation of one variable and the indirect manipulation of a second variable.

The three dimensional representation, similar to that shown in Figure 8.3, proposed by Heath (1969) of the variation of rate of photosynthesis with light intensity and carbon dioxide concentration, suggests an improvement in the display features which might encourage a perception of the importance of the interaction between variables. In the current version of *Bioview* a connected graph window displays a two dimensional rate/variable graph corresponding to the currently selected row or column. As an m_row/col or m_sheet operation is executed, the instance of this graph is replaced by another instance corresponding to the new row, column or sheet position. If each instance of the rate/variable graph was recorded for the execution of an uninterrupted sequence of m_row, m_col, or m_sheet operations, it would be possible to display a three dimensional graph representing the way the rate had changed with respect to changes in two variables. In fact this idea was suggested during the group discussion of the preliminary heuristic evaluation of *Bioview* (see Section 6.1.2). Figure 8.3 illustrates this idea with respect to either an uninterrupted sequence of the execution of m_row(C) operations on a T-sheet or an uninterrupted sequence of m_sheet(C) operations with a T-row highlighted. The figure shows line graphs, but a similar representation could be shown using bar graphs.

The direct manipulation of the datacube is very restricted; for example, it is not possible to manipulate the datacube itself to change row and column positions, and this would be an obvious improvement in the "directness" of the datacube design. As indicated above a major improvement in the directness of the datacube would result from the facility to directly manipulate the values of two or three variables at the same time. This would involve the simultaneous use of two operators to change two variables at the same time, that is, the simultaneous execution of (i) m_row and an m_col operations, (ii) m_row and m_sheet operations, or (iii) m-col and m_sheet operations. Changing three variables at the same time would involve the simultaneous execution of m_row, m_col, and m_sheet operations.

Double row-column manipulation could be effected by linked m_row and m_col operations. As the user moved from one cell position to another the associated row and column could be shown on the datasheet (and datacube), emphasising that the variation of two variables was being considered. This is illustrated in Figure 8.4 for a T-sheet.

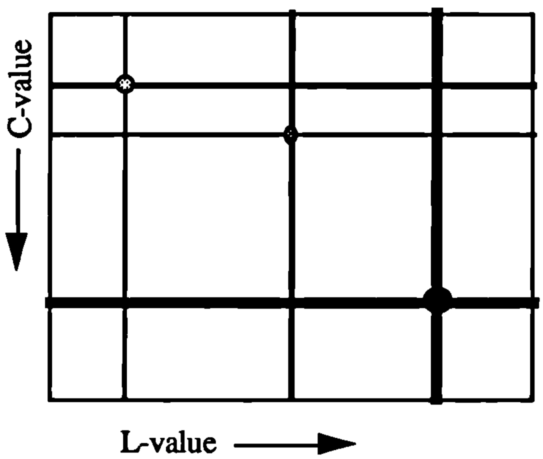


Figure 8.4: m_row and m_col double variable direct manipulation of the datasheet

The first position (light lines) corresponds to a low C-value and a low L-value. In the second position (medium lines) the value of light intensity has been increased, but the value of the carbon dioxide level has been further decreased. Both the light intensity and the carbon dioxide level have been increased in the third position (heavy lines). The rate graphs corresponding to these row and column positions are shown in Figure 8.5.

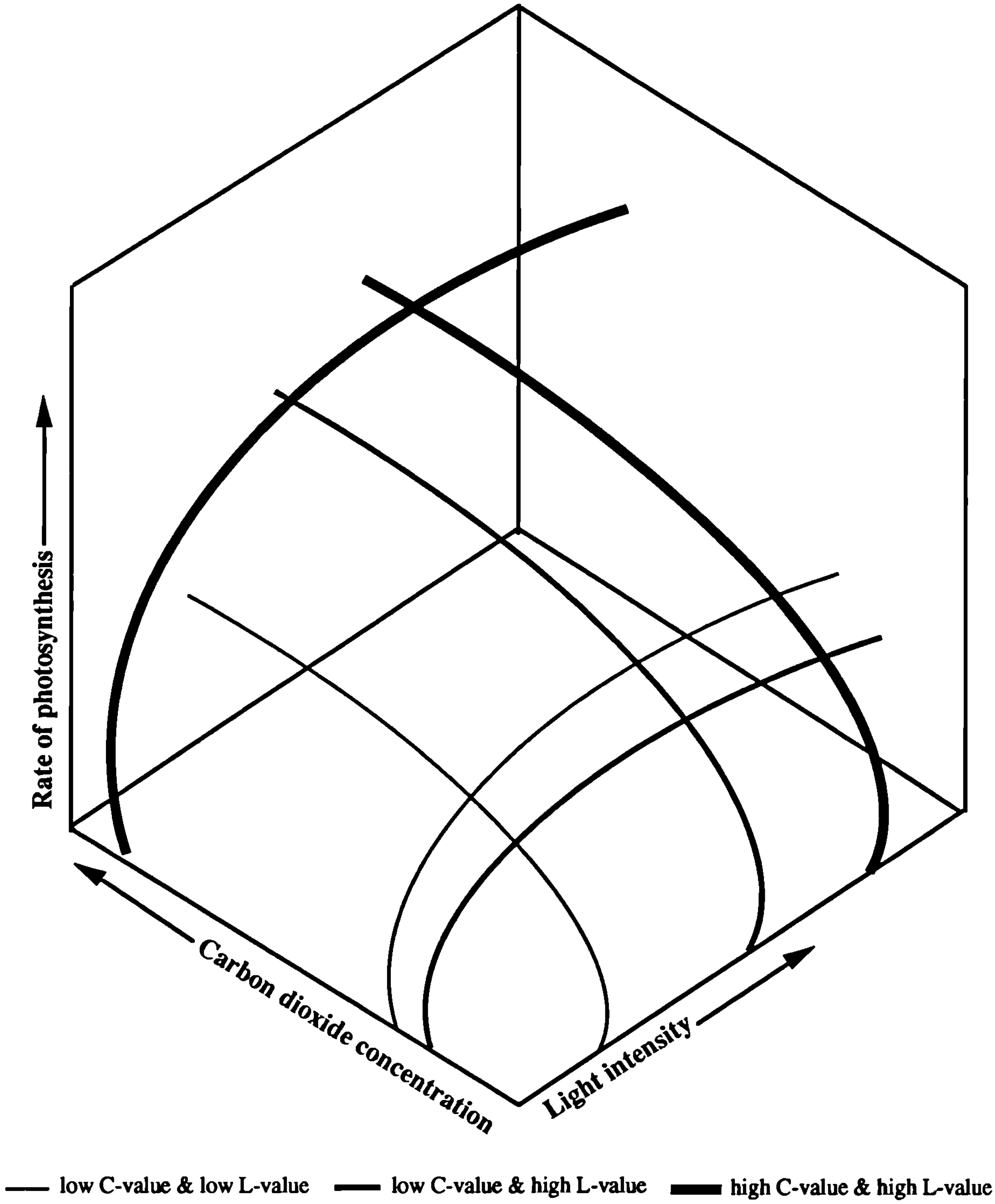


Figure 8.5: Representation of double variable direct manipulation of the datasheet

The design of direct manipulation features that would enable double (m-sheet, m_row/col) manipulation and triple (m_sheet, m_row, m_col) operations would present more problems. These could possibly be solved by the user depressing the shift-key whilst moving the mouse to indicate that the mouse movements should be interpreted in a three dimensional sense. A possible representation of triple variation is shown in Figure 8.6.

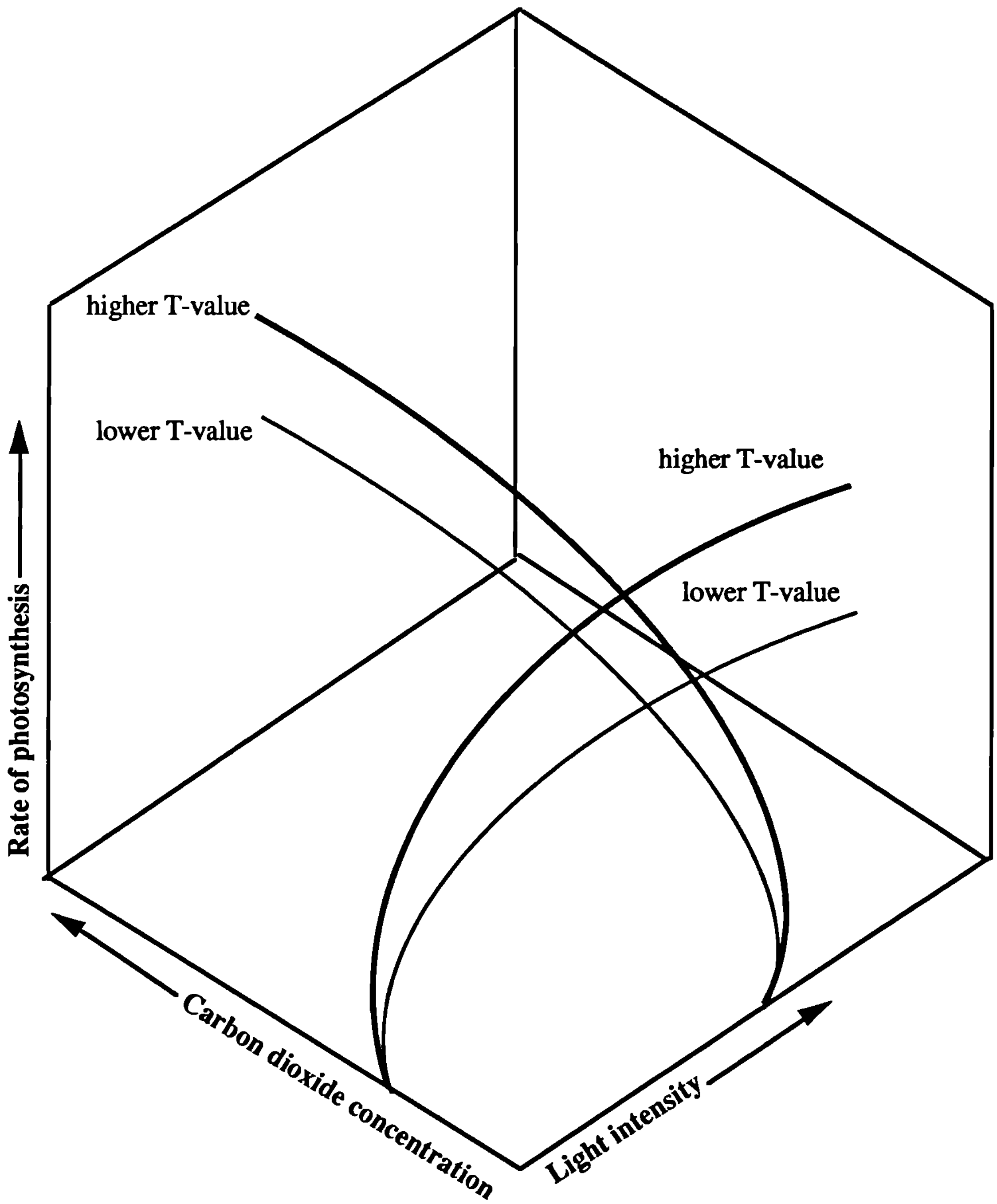


Figure 8.6: Representation of triple variable direct manipulation of the datasheet

From this critique of *Bioview* from a task perspective, it is clear that it is possible to combine aspects of the mitigation and design approaches advocated by Carroll & Roson (1987) to reduce the effect of the assimilation paradox.

8.3 Learner and designer models and direct manipulation educational software

The application of the Jigsaw Model in an evaluation of the use of *Bioview*, suggests that a comparison of learner and designer models can act as a basis for determining design guidelines for the development of direct manipulation software. Such a comparison leads to four design paradoxes which are discussed in the following sections.

8.3.1 The black box paradox

The black box paradigm for software design is based on the idea that the user does not need to understand how the system works. This is a superficially attractive doctrine - after all computers are complex artifacts requiring specialist knowledge to understand how they work. Surely one of the guiding principles of good design should be to hide this complexity from the user, thus removing any obligation to understand it. As Abowd & Beale (1991) propose, there should be a trade-off between user and system perceived difficulty in favour of the user:

> There is a trade-off between user and system perceived difficulty. A certain amount of work needs to be done to translate from the task language to the core language and back via the language of the interface. It is up to the designer of the system to decide whether the user or the system bears the brunt of the workload. We believe that it is a good policy to bias the interface heavily towards the user, thus minimising the user perceived difficulty. (Abowd & Beale, 1991, p. 79-80)

The black box approach is currently fashionable; interfaces are becoming more removed from the the system, indeed the Model World Metaphor for direct manipulation advocated by Hutchins et al. (1985, 1986) actively encourages users not to think about the system but to regard their interaction with the machine entirely in terms of the metaphor. In these terms the most successful metaphors provide a complete environment for the interaction - the metaphor *is* the system.

However, there is a price to pay for the inherent simplicity afforded to the user by the black box approach. The user is now completely reliant on the accuracy of the representation provided by the designer; there is no recourse to an explicitly described lower level model of the system with which to interpret problematical aspects of the interface performance. Even if the representation is accurate the learners have to assume that the designer's model of the system is the same, or at least consistent at some level, with the model of the system that they are using. Of course this may not be the case, as illustrated by the difference between the designer's and learners' surrogate models of *Bioview*; the designers model assumed a lexically scoped system register while the learner's model implied dynamic scoping of the datacube variables.

Thus, there is some scope for considering an approach to design which flies in the face of current conventional design wisdom - adopt the attack strategy described by Carroll & Rosson (1987) and do not hide a model of the system from the user, but make a representation of the model easily available to the learner. du Boulay et al. (1981) epitomise this approach in their concept of a "glass box" model of the computer as an aid to learning programming. The glass box model presents the user with a relatively simple model of the structure and function of the machine which is entirely adequate to understand the task in question. Providing a system register window in *Bioview* , as described in Section 8.2.3, provides another example of this approach.

This approach begs the question of at what level the system should be represented to the learner. Obviously a very low level representation close to the core language would in most cases be inappropriate and confusing. Representation at too high a level will result in a superficial model which will not be of genuine use, and may even be misleading. The short answer is that the representation should be at a level which ensures that the model formed by learners will be consistent with the designer's model. This does not necessarily mean that the designer and learner models should be the same; rather it means that there should be no differences between them which will cause misconceptions in the learner's model of the relationship between the interface and the system.

The discussion of consistency in Section 2.2 notes the debate over exactly what constitutes consistency. Grudin (1989) observed that consistency is usually defined in system terms and that this is not a good thing as it does not match users' requirements. The black box paradox supports his argument. If the user is unaware of the system model it is essential that the interface design is entirely consistent with the model of the task that they hold, as any unexpected results can only be reliably interpreted in terms of the knowledge of the task that they possess; they have no knowledge of the system to use as a yardstick.

The discussion of scoping in *Bioview* highlights the issue of the need for task based consistency in direct manipulation interfaces. Both lexical and dynamic scoping are consistent in a sense. Lexical scoping is consistent with respect to the system, and dynamic scoping is consistent with respect to the task. Conventionally, consistency tends to be interpreted in system terms, for example the use of lexical scoping in *Bioview*. However, as evidenced by the active system sub-register misconception, the significance of an interface which is consistent in system terms can be lost on the user, and may in fact induce problems. The disadvantages observed of adopting system based consistency, that is lexical scoping, as opposed to potential advantages of adopting a task based consistency, that is dynamic scoping, illustrate Grudin's point - good interface design utilises task related consistency.

Watson (1991), in her discussion of the development of *Scriptwriter* (a *Windows* based direct manipulation educational tool designed to support interactive reading and writing of stories), aptly highlights problems faced by adopting system based consistency rather than task based consistency:

> More fundamentally, it was a problem designing an interface which properly reflected the curriculum concerns of the educational designers rather than the functional concerns of the coders. Trials suggested that there is too heavy a dependence by the logic and language of rule making on the logic of the internal structure; this led to frustration by some users as the manipulation of the rules seemed too cumbersome. This supports the development experience which suggests that using powerful tools for implementation, and which carry a design rules of their own, places a burden on the developers not to allow system imperatives to override design considerations which must place educational considerations first. (Watson, 1991, p. 68-69)

Thus it seems that the black box paradox offers two design alternatives - if you make human-computer interaction simpler by representing the application as a black box do not make its use harder by making the interface consistent with the system; if you make human-computer interaction harder by requiring the user to have and possibly use a surrogate model of the system, make its use simpler by providing a clear representation of the system at an appropriate level.

8.3.2 The control paradox

The issue of learner control in the use of educational software is seen by many writers as a critical aspect of educational software design. Papert (1980) was an early advocate of the importance of learner control when he posited it as a fundamental principle to guide the conceptualisation of Logo as a programming language specifically designed for use in education; a theme echoed by Turkle (1983) in her notion of mastery in computer environments. Chandler (1984), Wellington (1985), McDougall & Squires (1986) and Blease (1986) all use the notion of learner control, as an organising theme in their discussion of frameworks for educational software. Underwood & Underwood (1990) claim that learner control is the single most important feature of the design of an educational software package. Thus giving the learner autonomy to choose aspects of application functionality, and the way in which they are used, is commonly regarded as a desirable feature of educational software. As such, learner control is conventionally regarded as a guiding principle in educational software design.

Learner control implies that the learner's model should take priority over the designer's model, which suggests that the designer's model is based on the perception of a malleable software environment which learners can alter to suit their own goals and methods. In this sense, the designer's model consists of a set of cognitive tools from which the user can choose. Necessarily, the designer must make value judgements as to what tools to provide; judgements which will be determined by the designer's model of the learner's model. At a fundamental level this model is based on the premise of the designer specifying the range and character of learning activities that can be supported by the software. However, within this context the model is predicated on the assumption that these cognitive tools are neutral artifacts that will be used by learners to pursue personal goals in their own ways. Therefore, there is an assumption that the designer's model of the learner's model recognises learners as purposeful active individuals who are able to take significant responsibility for their own learning.

Total delegation of control to users makes the learner entirely responsible for the way in which they use the software to support their learning. This assumes motivated, confident and experienced users. By definition, in an educational context, the user is a learner and these criteria are frequently not met in this setting, that is learners are typically not ready to assume full control of the software environment. Hence the notion arises of the designer as the learner's patron, with the designer assuming some control to mitigate the implications of total learner control. Carroll & Rosson (1987) describe this approach in the following terms:

> A second class of solutions moves the control up a level, so that the options available are controlled, rather than their consequences. So, for example, one can design or retrofit an interface so that so that advanced functions and/or potentially errorful trouble spots are unavailable to beginners (or more generally to users diagnosed as not ready for them). This is sometimes called a "staged" interface. Staging the presentation of a function can limit the range of errors that inexperienced users can fall into, and therefore make experimenting with the system less risky. (Carroll & Rosson, p. 88, 1987)

Mitigation through the delegation of control is a well established design principle for educational software. In its extreme form, delegation involves hiding aspects of the environment from the learner so that opportunities for the learner to exercise certain aspects of control simply do not arise. This approach is epitomised by the design rationale behind educational simulations. Typically only selected aspects of the functionality of the model underlying the simulation are revealed to the learner by the designer. As such, the designer is making value judgements as to what aspects of the exploratory environment offered by the simulation should be available for control by the learner.

Blocking, rather than hiding, some aspects of control from the learner is a less extreme form of delegation of control to the designer. The staged approach described by Carroll &

Rosson (1987) is based on blocking learner control. This approach is quite common in the design of educational software. For example, educational software often allows the learner to select from a range of difficulty levels, with each difficulty level incorporating learner control to a different extent. Another blocking technique is to make aspects of the environment evident but inaccessible to the learner. A specific example of this is provided by the design of the Star microcomputer interface (Smith, Erby, Kimbal, Verplank, & Harslem, 1982) which refrains from showing parameter options when an appropriate default value is available. Similar approaches are adopted in the design of educational software. For example, in some simulations the values of a set of parameters are displayed as the simulation is run, but it is impossible for the learner to change the values Another example of blocking is the provision of buried procedures in Logo microworlds - the learner is able to use these procedures but they are unable to change them.

The least extreme form of mitigation is effected by guiding the use of the software by the learner. The rationale behind this approach is to leave learners with maximum control, and to encourage them to exercise this control with caution. The use of default values provides a classic example of this approach. The default values enable learners to use software to produce intelligible and useful results, but they still have the freedom to change these values if they wish.

The issue of the delegation of learner control is critical to the direct manipulation paradigm. The success of a direct manipulation interface depends on the authenticity of the directness of the interface. Directness will only be authentic if the user feels totally in control of the software environment. If the designer patronises the learner by assuming some of this control the directness of the interface will be reduced. Thus, while there are compelling reasons for mitigating the effects of control by using techniques such as hiding and blocking functionality, the associated delegation of control to the designer can seriously compromise the validity of the direct manipulation that the use of the interface is predicated on. This dilemma is illustrated by the need for action confirmation observed during the use of *Bioview*. Learners had the freedom to produce null graphs, and in this sense they had total control over the system. However, as adduced by the number of consequent errors, this caused problems for the learners. As discussed in Section 8.2.2 mitigation could be achieved through hiding, blocking, or dimming rows and columns corresponding to sets of zero values. The price paid for the positive benefits of mitigation would be a compromise in the authenticity of the model world metaphor presented by the datacube, that is a compromise in directness.

Hence the control paradox: Direct manipulation typically gives learners extensive control but potentially exposes them to the perils of ambiguous and misleading feedback; mitigating the effects of this misleading feedback results in a compromise in the validity of direct manipulation.

8.3.3 The display paradox

It is now conventional wisdom that the display is an integral part of the user interface. This was not always the case, as illustrated by the screen design of early command driven interfaces. However, the widespread adoption of direct manipulation has required a conception of the role of the display in terms of the inter-referential nature of input and output (Draper, 1986). Thus appropriate display design is seen as an essential factor in the development of appropriate functional models of the system. Hence both the designer's and the learner's model will incorporate some consideration of the effect of the screen display, and the designer's model of the learner's model will need to account for any mismatch between the designer's and learner's perception of the display.

Making interaction easy in terms of a direct manipulation display implies efficient and economical manipulation of screen images. In turn, this implies that users should be required to make minimal changes to the display to achieve commonly desired effects. This implication is confirmed by the extensive display related truncation evident during the observations of the use of *Bioview*, and the fact that it was possible to identify display related selection rules that were used by the learners. The display related truncation observed during the use of *Bioview* indicates that the state of the display was instrumental in the selection of an appropriate method, that is the execution of operations by learners at the third and fourth levels of expertise identified by Black et al. (1987) were influenced by the state of the display. As discussed in Section 8.2.1 this influence is not always beneficial; it may be the case that the display "points" to the use of a method which is either sub-optimal or inappropriate, and the criterion for economical use has improperly constrained the use of the software.

While it is clear that the form and functionality of the screen display is an influential factor in human-computer interaction, it is not necessarily obvious at what level of expertise this influence is exercised. The learners observed using *Bioview* were operating at level three and above in terms of the expertise levels suggested by Black et al. (1987). For level two and one it is impossible for the screen to influence the selection of methods, as learners at this level do not employ methods. At the first level, the use of the system is based on preconceptions derived from experience gained prior to the use of the application, emphasising the distributed nature of user's models outlined by diSessa (1986). As such the implications of the display state will be assimilated by novice users in terms of this prior experience (Carroll & Rosson, 1987). In this sense the design features of the display will have a limited effect on the way learners use the software. However, at the second level the influence of the display may be more marked. At this level of initial learning simple functional models are formed in terms of links between operators and goals, making it probable that the display will be interpreted in terms of the selection of operators. Of course , it may be the case that quite different methods have the same initial operator. In such cases the learner may end up unconsciously making unsatisfactory method selections.

If a direct manipulation interface is going to help easy use of a software application, it should guide the user to form functional models which will enable operations to be efficiently and economically executed to achieve the desired effects. However, the most economical choice of operations may not correspond to the functional models held by experienced users. This is the first form of the display paradox. The second form relates to initial use of an application. Again the display should guide the learner in the choice of appropriate actions to execute, but at this level of expertise this guidance can only be given in simple terms which link operators directly to goals. Hence the second form of the display paradox - indicating the superficially appropriate choice of an operator may lead to the unconscious adoption of an inappropriate method. The designer's functional models will take cognisance of the potentially misleading influence of direct manipulation screen displays, but the learner's functional models will typically not do this. Thus the designer's model of the learner's model must recognise these misleading effects, taking account of the level of expertise of the learner.

8.3.4 The interaction paradox

As Norman (1991) pointed out, the introduction of a cognitive artifact will change the user's perception of the task that they are preforming through a distribution of cognition between the user and the artifact. Thus, while the actual task itself may remain the same, the responsibilities of the user will change. Norman used the simple example of a checklist to illustrate this point - without a checklist a person has to remember to execute a set of required operations; armed with a checklist the person no longer has this responsibility, and the requirement to remember the operations has been delegated to the cognitive artifact, that is the checklist. The way in which cognition is distributed between user and artifact will depend upon the nature of the interaction between user and artifact. Hence the design of the interface between artifact and user is critical in determining how the inherent functionality of the artifact affects the user's personal view of the task.

The comprehensive model world metaphor invoked by direct manipulation interfaces implies that the user's personal view of the functionality of a computer-based environment will be formed exclusively in terms of the direct manipulation techniques afforded by the interface. Therefore, the direct manipulation operators provided by the interface need to support effective management and manipulation of the interface and the specific requirements of the task in hand. Hence, the use of direct manipulation has metaphorical implications for both the computer-intrinsic and the IT-applicational domains identified by Birnbaum (1990).

The observed use of the direct manipulation of the datacube provided by *Bioview* illustrates how the nature of the direct manipulation techniques offered by an interface can affect the learner's view of a task. As discussed in Section 8.2.4, the value of only one variable can be varied at a time by *direct* manipulation, which may explain the use of an isolation of variables strategy by the learners, rather than the more appropriate multi-variable approach. Thus, while the datacube direct manipulation operators, (m_sheet and m_row/col operators) provided an intuitive and clear way of manipulating the datacube, they did not relate adequately to the task related requirements. In this sense, direct manipulation of the datacube successfully attended to the computer-intrinsic requirements of the learners, but failed to adequately address IT-applicational needs of the learners.

An interface that is successful in computer-intrinsic terms and relatively unsuccessful in task intrinsic terms indicates that there may be a conflict between designer and learner functional models. As noted earlier the designer's model is often based on a system based perception of consistency, while the learner's model tends to be formed in terms of the task requirements. This implies that the interface has been designed primarily with system related consistency in mind. Thus the persistent dilemma of system as opposed to task based consistency arises again, this time in the form of the interaction paradox: As the demand on the learner of coping with computer-intrinsic features is mitigated through direct manipulation consistent with the system, the direct manipulation techniques afforded by the interface tend to become progressively less task consistent, leading to the possibility of inappropriate application of the techniques.

8.4 Summary

The application of the Jigsaw Model to evaluate the cognition that occurred when *Bioview* was used to explore limiting factors in photosynthesis illustrates the comprehensive nature of the evaluation afforded by this model. The explanatory component of the model was able to accommodate an inner psychological theory relevant to the topic being studied, that is, a constructivist interpretation of learning, thus making it possible to utilise research into learning in this area. It was also possible to incorporate research and criticism relevant to the more general field of the use and understanding of interacting variables. In addition, proper regard was played to the influence that the system has in human-computer interaction. The indicative component demonstrated the possibility of expanding the GOMS approach to deal with a number of problematic areas: the consideration of non-expert performance, the commission of errors, software functionality, and individual differences.

The critique of *Bioview* based on the application of the Jigsaw Model was able to suggest a number of design improvements. These suggestions formed the background to a general consideration of the design of direct manipulation educational software in terms of the models held by the designer and the learner. It appears that learners need to have a surrogate model of the system in addition to workable functional models. However, the simple possession of a surrogate model is not sufficient in itself; it is critical that the surrogate model is consistent with the functional models that the learner holds. As the functional models relate to the features of the task domain, this implies that the surrogate model should be consistent in terms of task related criteria rather than system related criteria. If consistency is viewed primarily in system terms differences between learner and designer models result. The significance of such differences between learner and designer models is illustrated by four design paradoxes - the black box paradox, the control paradox, the display paradox, and the interaction paradox.

These paradoxes indicate design concerns which may need to be attended to in the design of high quality direct manipulation educational software. In addition, they suggest areas for further research. These suggestions need to be seen within the limitations of the research. The study was confined to an in-depth consideration of the use of one direct manipulation application in a well defined area of study, that is the use of *Bioview* to explore limiting factors in photosynthesis. All data collection in this phase took place within one day. Although the teacher-students had previously been introduced to *Bioview*, their familiarity with the package was necessarily limited. However, these data were supported by longitudinal data collected on other occasions, for example, questionnaire returns from Masters students and independent small scale empirical studies of the classroom use of *Bioview*. While extensive data in the form of audio transcripts, video records and interview transcripts, were collected during the major phase of the research, this phase was confined to three case studies. As such, the conclusions drawn are necessarily suggestive, rather than definitive.

The major contribution of the current research is the extension of the GOMS approach to include a comprehensive consideration of the cognitive aspects of the use of educational software, the Jigsaw Model. This model provides a framework for investigating each design paradox in the context of the use of other direct manipulation applications to support learning.

References

Abowd, G. D., & Beale, R. (1991). Users, Systems and Interfaces: A Unifying Framework for Interaction. In D. Diaper & N. Hammond (Eds.), *People and Computers VI*, (pp. 73-87). Heriot-Watt University: Cambridge University Press.

Adey, P., M, S., & Yates, C. (1989). *Thinking Science (The curriculum materials of the Cognitive Acceleration through Science Education (CASE) project).* Walton-on-Thames: Nelson.

Amir, R., & Tamir, P. (1989). When does a factor become a limiting factor? - a study of students' misconceptions. *Journal of Biological Education, 23*(2), 129-134.

Amir, R., & Tamir, P. (1990). Detailed analysis of misconceptions as a basis for developing remedial instruction: the case of photosynthesis. In *Annual Meeting of the American Educational Research Association,* (pp. 16-20). Boston, MA.

Amir, R., & Tamir, P. (1993). The "light" and "dark" reactions of photosynthesis: terminology as a source of misconceptions. In J. D. Novak (Ed.), *Third International Seminar on Misconceptions and Educational Strategies in Science and Mathematics.* Cornell University, Ithaca, New York: Department of Education, Cornell University.

Anderson, J. R. (1982). Acquisition of Cognitive Skill. *Psychological Review, 89*, 369-406.

Anderson, J. R. (1983). *The Architecture of Cognition.* Cambridge, MA: Harvard University Press.

Anderson, J. R. (1987). Skill acquisition: Compilation of weak-method problem solutions. *Psychological Review, 94*, 192-210.

Arnold, B., & Simpson, M. (1980). *An investigation of the development of photosynthesis to S. C. E. 'O' grade* , Aberdeen College of Education.

Barker, M., & Carr, M. (1989). Teaching and learning about photosynthesis. Part 1: An assessment in terms of students' prior knowledge. *International Journal of Science Education, 11*(1), 49-56.

Barnard, P. (1985). Interacting cognitive subsystems: A psycholinguistic approach to short term memory. In A. Ellis (Ed.), *Progress in the Psychology of Language* (pp. 197-258). London: Lawrence Erlbaum Associates.

Barnard, P. (1987). Cognitive resources and the learning of human-computer dialogues. In J. M. Carroll (Ed.), *Interfacing Thought: Cognitive Aspects of Human-Computer Interaction* (pp. 112-158). Cambridge, MA: MIT Press.

Barnard, P. J., & May, J. (1993). Cognitive Modelling for User Requirements. In P. F. Byerley, P. J. Barnard, & J. May (Eds.), *Computers, Communication and Usability: Research Methods for Integrated Services* (pp. 101-145). Amsterdam: Elsevier Science Publishers B. V.

Barnard, P. J., & Teasdale, J. (1991). Interacting Cognitive Subsystems: A systematic approach to cognitive affective interaction and change. *Cognition and Emotion, 5*(1), 5-39.

Barnard, P., Wilson, M., & MacLean, A. (1988). Approximate modelling of cognitive activity with an expert system: A theory based strategy for developing an interactive design tool. *The Computer Journal, 31*(5), 445-456.

Beech, G. (1983). *Computer based Learning: Practical Computing Methods*. Wilmslow: Sigma Technical Press.

Bell, B. (1985). Students' ideas about plant nutrition: What are they? *Journal of Biological Education, 19*(3), 213-218.

Bell, B., & Brook, A. (1984). *Aspects of secondary students' understanding of plant nutrition: full report*. Centre for the Study of Science and Mathematics Education, University of Leeds.

Bennett, J. L., Lorch, D. J., Kieras, D. E., & Polson, P. G. (1987). Developing a user interface technology for use in industry. In H. Bullinger & B. Shackel (Eds.), *Human-Computer Interaction - INTERACT 87*, (pp. 75-82). Elsevier Science Publishers B. V.

Birnbaum, I. (1990). The assessment of IT capability. *Journal of Computer Assisted Learning, 6*(1), 88-97.

Bitter, G., & Camuse, R. (1988). *Using a Microcomputer in the Classroom*. Englewood Cliffs, NJ: Prentice-Hall.

Black, J. B., Kay, D. S., & Soloway, E. M. (1987). Goal and Plan Knowledge Representations: From Stories to Text Editors and Programs. In J. M. Caroll (Ed.), *Interfacing Thought: Cognitive Aspects of Human Computer Interaction*. (pp. 36-60). Cambridge, MA: MIT Press.

Blease, D. (1988). *Evaluating Educational Software*. London: Croom Helm.

Bodker, S. (1989). A human activity approach to user interfaces. *Human-Computer Interaction, 4*, 171-195.

Bovair, S., Kieras, D., & Polson, P. G. (1990). The acquisition of and performance of text editing skill: A cognitive complexity analysis. *Human-Computer Interaction, 5*, 1-48.

Brook, A. (1985). Childrens' understanding of ideas about energy: A review of the literature. In R. Driver & R. Millar (Eds.), *Energy matters* (pp. 33 - 45). Centre for the Study of Studies in Science and Mathematics Education, University of Leeds.

Buxton, W. (1990). A three-state model for graphical input. In D. Diaper, D. Gilmore, G. Cockton, & B. Shackel (Eds.), *Human-Computer Interaction - INTERACT'90,* (pp. 449-456). North Holland.

Card, S. K., Moran, T. P., & Newell, A. (1980a). Computer text editing: An information-processing analysis of routine cognitive skill. *Cognitive Psychology, 12*, 32-74.

Card, S. K., Moran, T. P., & Newell, A. (1980b). The Keystroke-Level Model for User Performance Time with Interactive Systems. *Communications of the ACM, 23*(7), 396-410

Card, S. K., Moran, T. P., & Newell, A. (1983). *The Psychology of Human Computer Interaction.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Carroll, J. M., & Olson, J. R. (1988). Mental Models in Human-Computer Interaction. In M. Helander (Ed.), *Handbook of Human-Computer Interaction* (pp. 45-65). Amsterdam: Elsevier Science Publishers.

Carroll, J. M., & Rosson, M. B. (1987). Paradox of the Active User. In J. M. Carroll (Ed.), *Interfacing Thought: Cognitive Aspects of Human-Computer Interaction* (pp. 80-111). Cambridge, MA: MIT Press.

Carroll, J. M., & Thomas, J. C. (1988). Metaphor and the cognitive representation of computing systems. *IEEE Transactions on Systems, Man and Cybernetics, SMC-12*, 107-116.

Carroll, J. M., Kellogg, W. A., & Rosson, M. B. (1991). The Task-Artifact Cycle. In J. M. Carroll (Ed.), *Designing Interaction: Psychology at the Human-Computer Interface* (pp. 74-102). Cambridge: Cambridge University Press.

Chandler, D. (1984). *Young Learners and the Microcomputer.* Milton Keynes: Open University Press.

Dawson, C. J., & Rowell, J. A. (1986). All Other Things Equal: a study of science graduates solving control of variables problems. *Research in Science and Technological Education, 4*(1), 49-60.

diSessa, A. A. (1985). A principled design for an integrated computational environment. *Human-Computer Interaction, 1*, 1-47.

diSessa, A. A. (1986). Models of Computation. In D. A. Norman & S. W. Draper (Eds.), *User Centred System Design: New Perspectives on Human-Computer Interaction* (pp. 201-218). Hillsdale, NJ: Lawrence Erlbaum Associates.

diSessa, A. A. (1991). Local Sciences: Viewing the Design of Human-Computer Systems as Cognitive Science. In J. M. Carroll (Ed.), *Designing Interaction: Psychology at the Human-Computer Interface* (pp. 162-202). Cambridge: Cambridge University Press.

Dix, A., Finlay, J., Abowd, G., & Beale, R. (1993). *Human-Computer Interaction.* New York: Prentice Hall.

Draper, S. W. (1986). Display managers as the basis for user-machine communication. In D. A. Norman & S. W. Draper (Eds.), *User Centred System Design: New Perspectives on Human-Computer Interaction* (pp. 340-352). Hillsdale, NJ: Lawrence Erlbaum Associates Inc.

Driver, R., Child, D., Gott, R., Head, J., Johnson, S., Worsley, C., & Wylie, F. (1984). *Science in Schools at age 15* (Report to DES, DENI, and Welsh Office No. 2). Assessment of Performance Unit.

du Boulay, B., O'Shea, T., & Monk, J. (1981). The black box inside the glass box: presenting computing concepts to novices. *International Journal of Man-Machine Studies, 14*, 237-249.

Eberts, R. E., Kalappa, P., & Bittianda, P. (1993). Preferred mental models for direct manipulation and command-based interfaces. *International Journal of Man-Machine Studies, 38*, 769-785.

Eisen, Y., & Stavy, R. (1988). Students' understanding of photosynthesis. *The American Biology Teacher, 50*(4), 208-212.

Elkerton, J., & Palmiter, S. (1989). Designing help systems using a GOMS model: an information retrieval evaluation. *Human Factors, 33*, 185-204.

Finley, N. F., Stewart, J., & Yarroch, W. L. (1982). Teachers' perceptions of important and difficult science concepts. *Science Education, 66*(4), 531-538.

Fisher, R. A. (1960). *The Design of Experiments* (6th edition). Edinburgh: Oliver & Boyd.

Fitts, P. M., & Posner, M. I. (1967). *Human Performance*. Wadsworth Publishing.

Frese, M., Schulte-Gocking, H., & Altmann, A. (1987). Lernprozesse in Abhangigkeit von der Traingmethode, von Persdnmerkmalen und von der Benutzeroberflache (Direkte Manipulation vs. konventionelle interaktion). In W. Schonpflu & M. Wittstock (Eds.), *Software-Ergonomie '87* Berlin: Teubner.

Gaastra, P. (1962). Photosynthesis of Leaves and Field Crop. *Netherlands Journal of Agricultural Science, 10*(5), 311-324.

Gray, W. D., John, B. E., & Atwood, M. E. (1993). Project Ernestine: Validating a GOMS Analysis for Predicting and Explaining Real-World Task Performance. *Human-Computer Interaction, 8*, 237-309.

Green, N. P. O., Stout, G. W., & Taylor, D. J. (1990). *Biological Science 1 & 2*. Cambridge: Cambridge University Press.

Green, T. R. G., Schiele, F., & Payne, S. J. (1988). Formalisable Models of User Knowledge in Human Computer Interaction. In G. C. Van der Veer, T. R. G. Green, J.-M. Hoc, & D. Murray (Eds.), *Working with Computers: Theory versus Outcome* (pp. 3-46). London: Academic Press.

Grudin, J. (1989). The case against user interface consistency. *Communications of the ACM, 32*(10), 1164-1173.

Gugerty, L. (1993). The use of analytical models in human-computer-interface design. *International Journal of Man-Machine Studies, 38*, 625-660.

Halasz, F., & Moran, T. (1982). Analogy considered harmful. In *Conference on Human Factors in Computer Systems.* Gaithersburg, MD: National Bureau of Standards.

Haslam, F., & Treagust, D. F. (1987). Diagnosing secondary students' misconceptions of photosynthesis and respiration in plants using a two-tier multiple choice instrument. *Journal of Biological Education, 21*(3), 203-211.

Hayes-Roth, B., & Hayes-Roth, F. (1979). A Cognitive Model of Planning. *Cognitive Science, 3*, 275-310.

Heath, O. V. S. (1969). *The Physiological Aspects of Photosynthesis.* Stanford, CA: Stanford University Press.

Hill, T. (1991) *Microsoft Windows and Bioview: Changing the CAL emphasis in a changing user interface.* M. Ed., King's College London.

Hofmeister, A. (1984). *Microcomputer Applications in the Classroom.* New York: Holt, Rinehart and Winston.

Howes, A., & Payne, J. (1990). Display-based competence: towards user models for menu driven interfaces. *International Journal of Man-Machine Studies, 33*, 637-655.

Hutchins, E. L., Hollan, J. D., & Norman, D. A. (1985). Direct manipulation interfaces. *Human-Computer Interaction, 1*, 311-338.

Hutchins, E. L., Hollan, J. D., & Norman, D. A. (1986). Direct Manipulation Interfaces. In D. A. Norman & S. W. Draper (Eds.), *User Centred System Design: New Perspectives on Human-Computer Interaction* (pp. 87-124). Hillsdale, NJ: Lawrence Erlbaum Associates.

John, B. E. (1988) *Contributions to engineering models of human-computer interaction.* Ph. D., Carnegie Mellon University.

John, B. E. (1990). Extensions of GOMS Analyses to Expert Performance Requiring Perception of Dynamic Visual and Auditory Information. In J. Chew & J. Whiteside (Eds.), *Empowering People, Proceedings of the CHI'90 Conference*, (pp. 107-115). Seattle: ACM.

John, B. E., & Newell, A. (1989). Culminating the science of HCI: From S-R compatibility to transcription typing. In K. Bice & C. Lewes (Eds.) *Conference on Human Factors in Computing Systems - CHI'89*, (pp. 109-114). Austin: ACM.

Johnstone, A. H., & Mahmoud, N. A. (1980). Isolating topics of high perceived difficulty in school biology. *Journal of Biological Education, 14*(2), 163-166.

Jungwirth, E. (1988). Comments on Wandersee's article "Can the history of science help science educators anticipate students' misconceptions". *Journal of Research in Science Teaching*, *25*(2), 161-162.

Karat, J., Fowler, R., & Gravelle, M. (1987). Evaluating user interface complexity. In H. Bullinger & B. Shackel (Eds.), *Human-Computer Interaction - INTERACT'87*, (pp. 189-196). Elsevier Science Publishers B. V.

Karrat, J., & Bennett, J. (1989). *Modelling the user interaction methods imposed by designs* (Report No. RC 14649 (#65149)). IBM, Watson Research Centre.

Kay, D. S., & Black, J. B. (1984). The changes in knowledge representations of computer systems with experience. In *Proceedings of the Human Factors Society 28th Annual Meeting*. Santa Monica: Human Factors Society.

Kay, D. S., & Black, J. B. (1986). *Knowledge transformations during the acquisition of computer expertise* (CCT Report No. 86-2). New York Teachers College, Columbia University.

Kellog, W. A. (1987). Conceptual consistency in the user interface: effects on user performance. In H. J. Bullinger & B. Shackel (Eds.), *Human-Computer Interaction - INTERACT'87*, (pp. 389-394). Elsevier Science Publishers B. V.

Kemmis, S., Atkin, R., & Wright, E. (1977). *How Do Students Learn?* (Occasional Paper No. 5). Centre for Applied Research in Education, University of East Anglia.

Kieras, D. E. (1988). Towards a Practical GOMS Model Methodology for User Interface Design. In M. Helander (Ed.), *Handbook of Human-Computer Interaction* (pp. 135-157). Amsterdam: Elsevier Science Publishers B. V.

Kieras, D., & Bovair, S. (1986). The acquisition of procedures from text: A production system analysis of transfer of training. *Journal of Memory and Language*, *25*, 507-524.

Kieras, D., & Polson, P. G. (1985). An approach to the formal analysis of user complexity. *International Journal of Man-Machine Studies*, *22*, 365-394.

Laird, J. F., Newell, A., & Rosenbloom, P. (1987). Soar: an architecture for general intelligence. *Artificial Intelligence*, *33*(1), 1-64.

Lee, A. Y., Polson, P. G., & Bailey, W. (1985). Learning and transfer of measurement tasks. In K. Bice & C. Lewes (Eds.), *Conference on Human Factors in Computing Systems - CHI'89*, (pp. 115-120). ACM.

Lerch, F. J. (1988) *Computerised financial planning: Discovering cognitive difficulties in model building*. Ph. D., University of Michigan, Ann Arbor.

Lerch, F. J., Mantei, M. M., & Olson, J. R. (1989). Translating ideas into action: Cognitive analysis of errors in spreadsheet formulas. In K. Bice & C. Lewes (Eds.) *Conference on Human Factors in Computing Systems - CHI'89*, (pp. 121-126). ACM.

Lewis, C. (1991). Inner and Outer Theory in Human-Computer Interaction. In J. M. Carroll (Ed.), *Designing Interaction: Psychology at the Human-Computer Interface* (pp. 154-161). Cambridge: Cambridge University Press.

Lucas, A. M., & Tobin, K. (1987). Problems with "Control of Variables" as a Process Skill. *Science Education, 71*(5), 685-690.

Malone, T. W. (1981). Towards a Theory of Intrinsically Motivating Instruction. *Cognitive Science, 4*, 333-369.

Mayes, J. T., Draper, S. W., McGregor, M. A., & Oatley, K. (1988). Information flow in a user interface: the effect of experience and context on the recall of MacWrite screens. In D. M. Jones & R. Winder (Eds.), *People and Computers IV*, (pp. 275-289). Cambridge: Cambridge University Press.

McDougall, A., & Squires, D. (1986). Student Control in Computer Based Learning Environments. In A. D. Salvas & C. Dowling (Eds.), *Computers in Education: On the Crest of a Wave*, (pp. 269-272). Melbourne: Computer Education Group of Victoria.

Moran, T. P. (1981). Command language grammar. *International Journal of Man-Machine Studies, 15*, 3-50.

Nelson, T. (1980). Interactive systems and the design of virtuality. *Creative Computing, 6*(11), 35-38.

Newell, A., & Simon, H. A. (1972). *Human Problem Solving*. Englewood Cliffs, NJ: Prentice Hall.

Newman, J. (1988). Software Libraries: the Backbone of Schools Computing. In P. Alp (Ed.), *Australian Computer Education Conference*, (pp. 242-251). Perth: Educational Computing Association of Western Australia.

Nielsen, J., & Molich, R. (1990). Heuristic Evaluation of User Interfaces. In J. Chew & J. Whiteside (Eds.), *Empowering People, Proceedings of the CHI'90 Conference*, (pp. 249-256). Seattle: ACM.

Norman, D. A. (1983). Some observations on mental models. In D. Getner & A. Stevens (Eds.), *Mental Models* (pp. 7-14). Hillsdale, NJ: Lawrence Associates Associates.

Norman, D. A. (1991). Cognitive Artifacts. In J. M. Carroll (Ed.), *Designing Interaction: Psychology at the Human-Computer Interface* (pp. 17-38). Cambridge: Cambridge University Press.

Norman, D. A. (1986). Cognitive Engineering. In D. A. Norman & S. W. Draper (Eds.), *User Centred System Design* (pp. 31-61). Hillsdale, NJ: Lawrence Erlbaum Associates.

Nuffield Advanced Science (1985). *Biology Study Guide 1*. Harlow: Longman.

OECD (1988). *Information Technologies in Education: The Quest for Quality Software*. Paris: Organisation for Economic Co-operation and Development.

Olson, J. R., & Nilsen, E. (1988). Analysis of the cognition involved in spreadsheet software interaction. *Human-Computer Interaction, 3*, 309-350.

Olson, J. R., & Olson, G. M. (1990). The Growth of Cognitive Modelling in Human-Computer Interaction Since GOMS. *Human-Computer Interaction, 5*(2&3), 221-265.

O'Shea, T., & Self, J. (1983). *Learning and Teaching with Computers*. Brighton: The Harvester Press.

OTA (1988). *Power On! New Tools for Teaching and Learning*. Washington, D.C: US Government Printing Office.

Papert, S. (1980). *Mindstorms: Children, Computers and Powerful Ideas*. Brighton: Harvester Press.

Payne, S. J. (1987). Complex problem space: modelling the knowledge needed to use interactive devices. In H. J. Bullinger & B. Shackel (Eds.), *Human-Computer Interaction - INTERACT'87*, (pp. 203-208). Elsevier Science Publishers B. V.

Payne, S. J. (1991). Interface Problems and Interface Resources. In J. M. Carroll (Ed.), *Designing Interaction: Psychology at the Human-computer Interface* (pp. 128-153). Cambridge: Cambridge University Press.

Payne, S. J., & Green, T. R. G. (1983). The user's perception of the interaction language: A two level model. In *Conference on Human Factors in Computer Systems - CHI'83*, (pp. 202-206). ACM.

Payne, S. J., & Green. T. R. G. (1986). Task-Action Grammars: A Model of the Mental Representation of Task Languages. *Human-Computer Interaction, 2*, 93 -133.

Pelgrum, J., & Plomp, T. (1991). *The Use of Computers Worldwide*. Oxford: Pergamon Press.

Phillips, W. D., & Chilton, T. J. (1989). *A-Level Biology*. Oxford: Oxford University Press.

Polson, P. G. (1987). A quantitative model of human-computer interaction. In J. M. Carroll (Ed.), *Interfacing thought: Cognitive aspects of human-computer interaction* (pp. 184-235). Cambridge, MA: MIT Press.

Polson, P. G., & Kieras, D. E. (1985). A quantitative model of the learning and performing of text editing knowledge. In L. Borman & B. Curtis (Eds.) *Conference on Human Factors in Computing Systems - CHI'85*, (pp. 207-212). San Francisco: ACM.

Polson, P. G., Muncher, E., & Englebeck, G. (1986). A test of common elements theory of transfer. In *Conference on Human Factors in Computing Systems - CHI'86*, (pp. 78-83). ACM.

Rabinowitch, E. I. (1951). *Photosynthesis and related processes*. New York: Interscience.

Rabinowitch, E. I. (1956). *Photosynthesis and related factors*. New York: Interscience.

Reisner, P. (1981). Formal Grammar and the Design of an Interactive System. *IEEE Transactions on Software Engineering, 7*, 229-240.

Reisner, P. (1993). APT: a description of user interface inconsistency. *International Journal of Man-Machine Studies, 39*, 215-236.

Riley, D. (1991). *Bioview Users' Guide*. London: King's College London.

Roberts, M. B. V. (1981). *Biology for Life*. Walton-on-Thames: Nelson.

Roth, K. J., Smith, E. L., & Anderson, C. W. (1983). Students' conceptions of photosynthesis and food for plants. In *Annual Meeting of the American Educational Research Association*. Montreal.

Rowntree, D. (1982). *Educational Technology in Curriculum Development*. London: Harper and Row.

Rubinstein, R., & Hersh, H. (1984). *The human factor, designing computer systems for people*. Burlington, VT: Digital Press.

Rutkowski, C. (1982). An introduction to the Human Applications Standard Computer Interface, Part 1: Theory and Principles. *Byte, 7*(11), 291-310.

Salvas, A. G., & Thomas, G. J. (1984). *Evaluation of Software*. Melbourne: Education Department of Victoria.

Sebrechts, M. M., Black, J. B., Galambos, J. A., Wagner, R. K., Deck, J. A., & Wikler, E. A. (1983). *The effects of diagrams on learning to use a system* (Learning and Using Systems Project Report No. 2). Yale University.

Self, J. (1985). *Microcomputers in Education: A Critical Appraisal of Educational Software*. Brighton: Harvester Press.

Shayer, M. (1986). Data processing and science investigation in schools. *Research Papers in Education, 1*(3), 237-253.

Shneiderman, B. (1982). The future of interactive systems and the emergence of direct manipulation. *Behaviour and Information Technology, 1*(3), 237-256.

Shneiderman, B. (1983). Direct Manipulation: A Step Beyond Programming Languages. *IEEE Computer, 16*, 57-69.

Shneiderman, B. (1987). *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. New York: Addison-Wesley.

Shneiderman, B., & Margono, S. (1987). A Study of File Manipulation by Novices Using Commands vs. Direct Manipulation. In *26th Annual Technical Symposium of the Washington, D.C. Chapter of the ACM*. Gathersburgh, MD: National Bureau of Standards.

Simon, T. (1988). Analysing the Scope of Cognitive Models Human-Computer Interaction: A Trade-Off Approach. In D. M. Jones & R. Winder (Eds.), *People and Computers IV*, (pp. 79-93). University of Manchester: Cambridge University Press.

Simonson, R., & Thompson, A. (1990). *Educational Computing Foundations*. New York: MacMillan.

Simpson, M., & Arnold, B. (1982a). The inappropriate use of subsumers in biology learning. *European Journal of Science Education, 4*(2), 173-182.

Simpson, M., & Arnold, B. (1982b). Availability of prerequisite concepts for learning biology at certificate level. *Journal of Biological Education, 16*(1), 65-72.

Singley, M. K., & Anderson, J. R. (1988). A keystroke analysis of learning and transfer in text editing. *Human-Computer Interaction, 3*, 223-274.

Smelcer, J. B. (1988) *Understanding user errors in database query*. Ph. D., University of Michigan.

Smith, D., Irby, C., Kimbal, R., Verplank, B., & Harslem, E. (1982). Designing the Star Interface. *Byte, 7*(4), 242-282.

Smith, E. L., & Anderson, C. W. (1984). Plants as producers, a case study of elementary science teaching. *Journal of Research in Science Teaching, 21*(7), 685-698.

Soyibo, K. (1983). Selected science misconceptions among some Nigerian school certificate students. In H. Helm & J. D. Novak (Eds.), *First International Seminar on Misconceptions and Educational Strategies in Science and Mathematics*, (pp. 425-427). Cornell University, Ithaca, New York: Department of Education, Cornell University.

Springett, M. V. (1992). The Utility of User Action Models for Direct Manipulation Design. In J. Larson & C. Unger (Eds.), *IFIP TC2/WG2.7 Working Conference on Engineering for Human Computer Interaction*, (pp. 205-222). Elsevier Science Publishers B. V.

Springett, M. V., & Grant. A. S. (1993). Interface Semantics and Users' Device Models: Identifying Evaluation Issues for Direct Manipulation Design. In J. L. Alty, D. Diaper, & S. Guest (Eds.), *People and Computers VIII*, (pp. 249-265). Loughborough: Cambridge University Press.

Squires, D. (Ed.). (1991). *Bioview*. London: King's College London.

Squires, D., & McDougall, A. (1994). *Choosing and Using Educational Software: A Teachers' Guide*. London: Falmer Press.

Stavy, R., Eisen, Y., & Yaakobi, D. (1987). How students aged 13-15 understand photosynthesis. *International Journal of Science Education, 9*(1), 105-115.

Sutcliffe, M. V., & Springett, A. G. (1992). From Users' Problems to Design Errors: Linking Evaluation to Improved Design Practice. In A. Monk, D. Diaper, & M. Harrison (Eds.), *People and Computer VII*, (pp. 117-131). Cambridge University Press.

Tauber, M. J. (1986a). An approach to metacommunication in human-computer interaction. In F. Klix & H. Wandke (Eds.), *Man-Computer Interaction Research MACINTER1* (pp. 35-50). Amsterdam: North Holland.

Tauber, M. J. (1986b). Top-down design of human-computer interaction. In S. K. Chang, T. Ichikawa, & P. A. Ligomenides (Eds.), *Visual Languages* (pp. 289-308). New York: Plenum Press.

Tauber, M. J. (1988). On Mental Models and the User Interface. In G. C. van der Veer, T. R. G. Green, J.-M. Hoc, & D. Murray (Eds.), *Working with Computers: Theory versus Outcome* (pp. 89-119). London: Academic Press.

Taylor, R. P. (Ed.). (1980). *The Computer in School: Tutor, Tool, Tutee*. New York: Teachers College Press.

Teasdale, J., & Barnard, P. J. (1993). *Affect, Cognition and Change: Re-modelling depressive thought*. Hove: Lawrence Erlbaum Associates.

Thompson, B. (1991) *An Evaluation of Bioview*. M. Ed., King's College London.

Toole, G., & Toole, S. (1991). *Biology for Advanced Level* (Second ed.). Cheltenham: Stanley Thornes.

Turkle, S. (1983). *The Second Self: Computers and the Human Spirit*. New York: Simon and Schuster.

Underwood, J. D. M., & Underwood, G. (1990). *Computers and Learning*. Oxford: Blackwell.

Vossen, P. H., Sitter, S., & Ziegler, J. E. (1987). An empirical validation of cognitive complexity theory with respect to text, graphics and table editing. In H. J. Bullinger & B. Shackel (Eds.), *Human-Computer Interaction - INTERACT'87*, (pp. 309-316). Elsevier Science Publishers B. V.

Waheed, T., & Lucas, A. M. (1992). Understanding interrelated topics: photosynthesis at age 14+. *Journal of Biological Education, 26*(3), 193-199.

Wandersee, J. H. (1983). Students' misconceptions about photosynthesis: a cross age study. In H. Helm & J. D. Novak (Eds.), *First International Seminar on Misconceptions and Educational Strategies in Science and Mathematics* (pp. 441-446). Cornell University, Ithaca, New York: Department of education, Cornell University.

Wandersee, J. H. (1985). Can the history of science help science educators anticipate students' misconceptions. *Journal of Research in Science Teaching, 23*(7), 581-597.

Wandersee, J. H. (1988). Author's response to Ehud Jungwirth. *Journal of Research in Science Teaching, 25*, 163-164.

Watson, D. (1991). Scriptwriter: Framework software and its relation to the curriculum. *Computers and Education, 16*(1), 65-69.

Wellington, J. J. (1985). *Children, Computers and the Curriculum*. New York: Harper and Row.

Whiteside, J., & Wixon, D. (1987). Discussion: improving human-computer interaction - a quest for cognitive science. In J. M. Carroll (Ed.), *Interfacing Thought: Cognitive Aspects of Human-Computer Interaction* (pp. 353-365). Cambridge, MA: MIT Press.

Whiteside, J., Jones, S., Levy, P., & Wixon, D. (1985). User Performance with Command, Menu and Iconic Interfaces. In L. Borman & B. Curtis (Eds.) *Conference on Human Factors in Computer Systems - CHI'85*, (pp. 185-191) San Francisco: ACM.

Wood-Robinson, C. (1991). Young peoples' ideas about plants. *Studies in Science Education, 19*, 119-135.

Wright, P., Monk, A. F., & Carey, T. (1991). *Cooperative Evaluation - The York Manual*. Department of Psychology, University of York.

Young, R. (1981). The machine inside the machine: users' models of pocket calculators. *International Journal of Man-Machine Studies, 15*, 51-85.

Young, R. M. (1983). Surrogates and mappings: Two kinds of conceptual models for interactive devices. In D. Getner & A. L. Stevens (Eds.), *Mental Models* (pp. 35-52). Hillsdale, NJ: Lawrence Erlbaum Associates.

Young, R. M., & Green, T. R. G. (1989). Programmable user models for predictive evaluation of interface designs. In K. Bice & C. Lewis (Eds.), *Human Factors in Computing Systems - CHI'89*, (pp. 15-19). ACM Press.

Ziegler, J. E., & Fahnrich, K. P. (1988). Direct Manipulation. In M. Helander (Ed.), *Handbook of Human-Computer Interaction* (pp. 123-133). Amsterdam: Elsevier Science Publishers B. V.

Appendix 1

A description of *Bioview*

This appendix provides a full description of *Bioview*. In Section 1 the windows that can be opened in *Bioview* are described. The links between these windows are discussed in Section 2. The terminology for *Bioview* operators introduced in Chapter 4 is used in this appendix.

Bioview windows

A typical screen display for *Bioview* is shown in Figure A1.1.

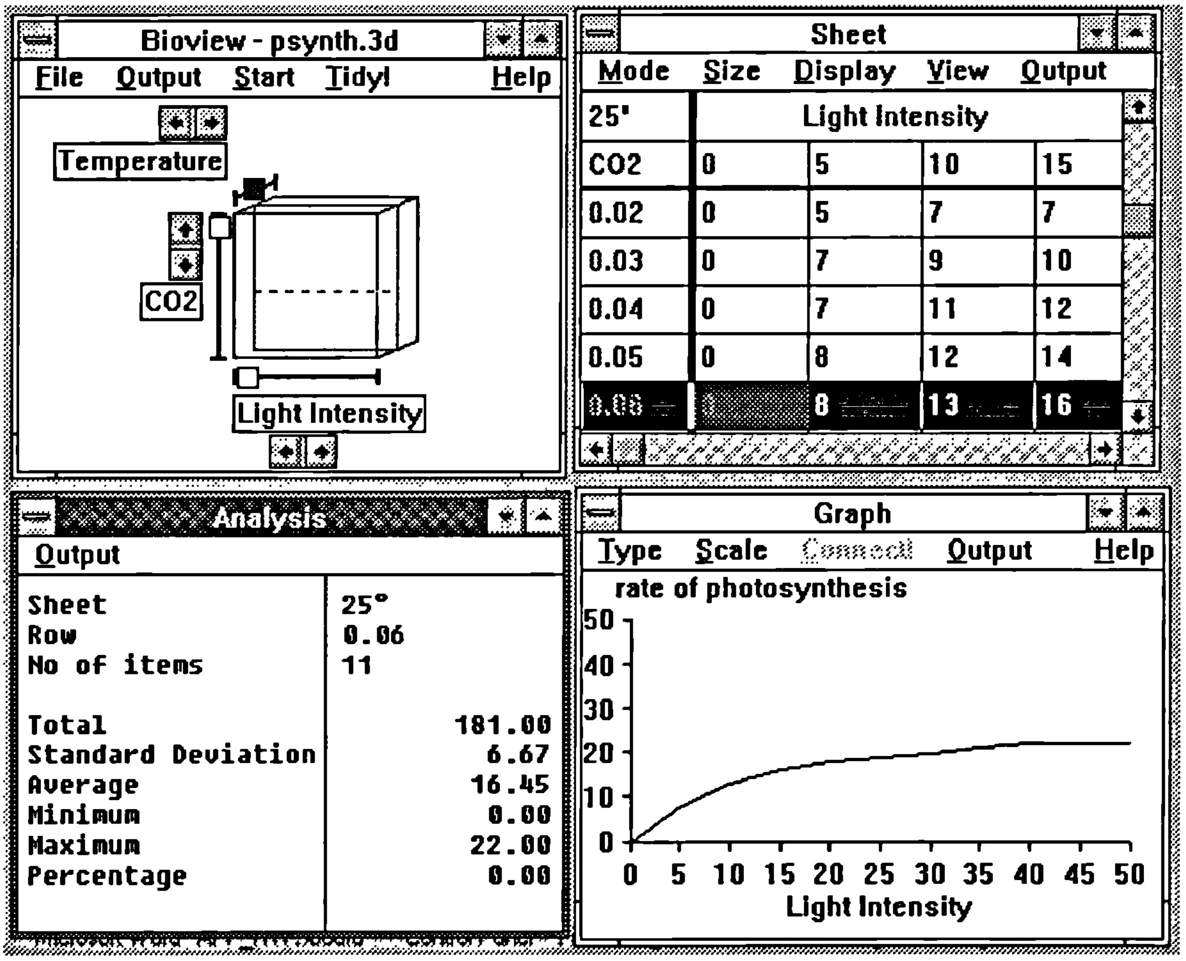


Figure A1.1: A representative Bioview screen display

The four types of window that it is possible to open are shown: a datacube window, a datasheet window, a graph window, and an analysis window. As with any *Windows* application these windows can be moved and resized to overlap with each other, maximised to fill the whole screen, and minimised as an icon. Each type of window is discussed below.

The datacube window

The datacube window (top left hand portion of the screen display) presents an image of the datacube which can be directly manipulated either to select a datasheet or move the position of the currently selected datasheet. The position of a sheet can be changed by either clicking on or dragging one of the three value scroll boxes associated with the datacube edges. Clicking on one of these boxes will result in moving the position of the sheet one step at a time to positions corresponding to each defining sheet value. For example, with the PSYNTH datacube this means moving the sheet to five successive positions corresponding to T-values of 10°C, 20°C, 25°C, 30°C, and 40°C. In Figure A1.1 the sheet has been moved to the position corresponding to 25°C. Dragging the box along the slider will produce a rapid change in sheet positions and corresponding sheet values. A different sheet can be selected by either clicking on a different value scroll box or the appropriate datacube face. For example, to select a sheet defined by a fixed value for the level of carbon dioxide either the value scroll box corresponding to the carbon dioxide level or the top face of the datacube should be clicked on. The currently selected row or column on the datasheet is indicated by a dotted line. In Figure A1.1 a row is selected.

The datacube window menu bar provides access to three program management options and one screen management option. The program management options are Open, Output and Start, and the screen management option is Tidy! Selecting the Open menu item produces a drop down menu from which it is possible to load a datacube, save the currently loaded datacube, name and save the currently loaded datacube, create a new datacube, set the size of the datacube, and label the datacube axes and cells. Output enables a copy of the datacube to be pasted to the *Windows* clipboard or printed. The Start option allows graph windows to opened. This option also allows the *Write* and *Notepad Windows* applications to be loaded. It is possible to have multiple graph windows selected, consistent with the memory constraints of the host machine. Selecting Tidy! results in the datacube, datasheet, initial graph, and analysis windows defaulting to their original positions (as shown in Figure A1.1).

The datasheet window

The datasheet window is shown in the top right hand portion of the screen. If a row is highlighted the position of the row can be moved by clicking on a new position or dragging the row a new position. Clicking will result in the movement from a row corresponding to one variable value directly to a row corresponding to another variable value. For example, clicking on a new row position for a T-sheet will result in a change in carbon dioxide concentration values directly from one C-value to another C-value. Dragging the row position will result in the selection of a succession of C-values. A column position can also be changed by clicking and dragging in the same way. The top menu bar in the datasheet window provides a Display option which allows rows or columns to be selected.

The datasheet window only displays as much of the datasheet as will fit into the window. The extent of the data displayed can be increased in four ways. Firstly, the datasheet window can be maximised to fill the whole screen. This has the advantage of showing the maximum amount of data, but, of course, all the other windows are obscured. Secondly, the Size option in the top menu bar can be selected to change the column width of the display. Thirdly, the datasheet window can be resized. Fourthly, the window scroll boxes (one in each corner of the window) can be used to horizontally or vertically scroll through the data, thus changing the scope of the window. This fourth option results in the execution of scope_win operators.

View provides an alternative to the use of s_sheet operators. Selecting this option from the top menu bar produces a drop down menu from which it is possible to select one of the three datasheets for display in the datasheet window. In this way one aspect of the direct manipulation of the datacube, that is the selection of a datasheet, can be replaced by a menu selection procedure. However, this is the only direct manipulation technique in *Bioview* which can be avoided in this fashion.

It is possible to copy the datasheet to the *Windows* clipboard or print the datasheet currently shown on the screen by choosing from the drop down menu produced when the Output option is selected from the top menu bar in the datasheet window. In addition a "Print all" menu item allows all the datasheets corresponding to the currently selected "slice" of the datacube to be printed.

Bioview can be used with existing datacubes or new datacubes can be created by the user. It is also possible to edit existing datacubes. The Mode option in the top menu bar allows the user to choose between data entry and display. If data display is selected the data in the spreadsheet are locked, and it is not possible to edit data. Selection of data entry allows changes to the data to be made, that is new data can be typed in and existing data can be changed by clicking on the appropriate datasheet cells and typing in new values.

Graph windows

The default initial screen configuration has one graph window open and located in the bottom right hand corner of the screen. As explained earlier further graph windows can be opened by using the Start option in the top menu bar of the datacube window. These graph windows can be located in any part of the screen, but a common arrangement is as shown in Figures A1.2 and A1.3 with two graph windows of roughly equal size displayed side by side. Only one graph can be simultaneously connected to the datacube and datasheet windows at a time, that is the effects of directly manipulating the datasheet and datacube can only be reflected in the instance of a rate/variable graph shown in one graph window. Clicking on the Connect option in the top menu bar of the graph window will connect the currently shown graph to the datacube and datasheet.

Selecting the Type menu item produces a drop down menu with three items: line graph, bar chart, and pie chart. Choosing one of these items will result in the display of the chosen type of graph. It is not necessary for the graph window to be connected to change the type of graph displayed in the graph window. A graph can be scaled by choosing the Scale option from the top bar in the appropriate graph window. It is possible to scale the currently shown graph with respect to the maximum value in the row or column corresponding to he current graph, the maximum value in the currently selected datasheet, or the maximum value in the datacube. If an m_row or an m_col operation is attempted when row or column based scaling has been chosen, the scaling will automatically be changed to datasheet based scaling. Similarly, if an m_sheet operation is attempted when scaling based either on a sheet or a row or column has been selected, the scaling will automatically be changed to datacube based scaling. The Output option allows a copy of a graph window to be pasted into the *Windows* clipboard or printed.

The version of *Bioview* used in the laboratory sessions showed the values of the currently selected sheet and row or column values at the top of each graph window.

The analysis window

This window provides a simple statistical summary of the currently selected row or column. An example of an analysis window is shown in the bottom left hand corner of the screen display shown in Figure A1.1. The defining values of the datasheet and row or column are displayed. In Figure 1 these are shown as 25°C for the sheet, and a row value of 0.06. The number of items in the C-row is shown as 11. The total, standard deviation, average value, minimum value, and maximum value are displayed for the currently selected row or column as shown in Figure A1.1. In addition, the percentage of a specific datum value with respect to the total is shown. The specific datum item is selected by clicking on the appropriate cell in the highlighted row or column.

Unlike graph windows it is only possible to display one analysis window. This window is automatically connected to the datacube and datasheet, that is the results of manipulating the datacube and the datasheet are always reflected in changes in the values displayed in this window. The Output option in the top menu bar enables the analysis window to be pasted to the *Windows* clipboard or printed.

Links between *Bioview* windows

Figure A1.2 illustrates the execution of an m_sheet operation. The T-sheet has been moved from an intermediate T-value of 25°C to the maximum T-value of 40°C. The row position, corresponding to a C-value of 0.06, has not been changed.

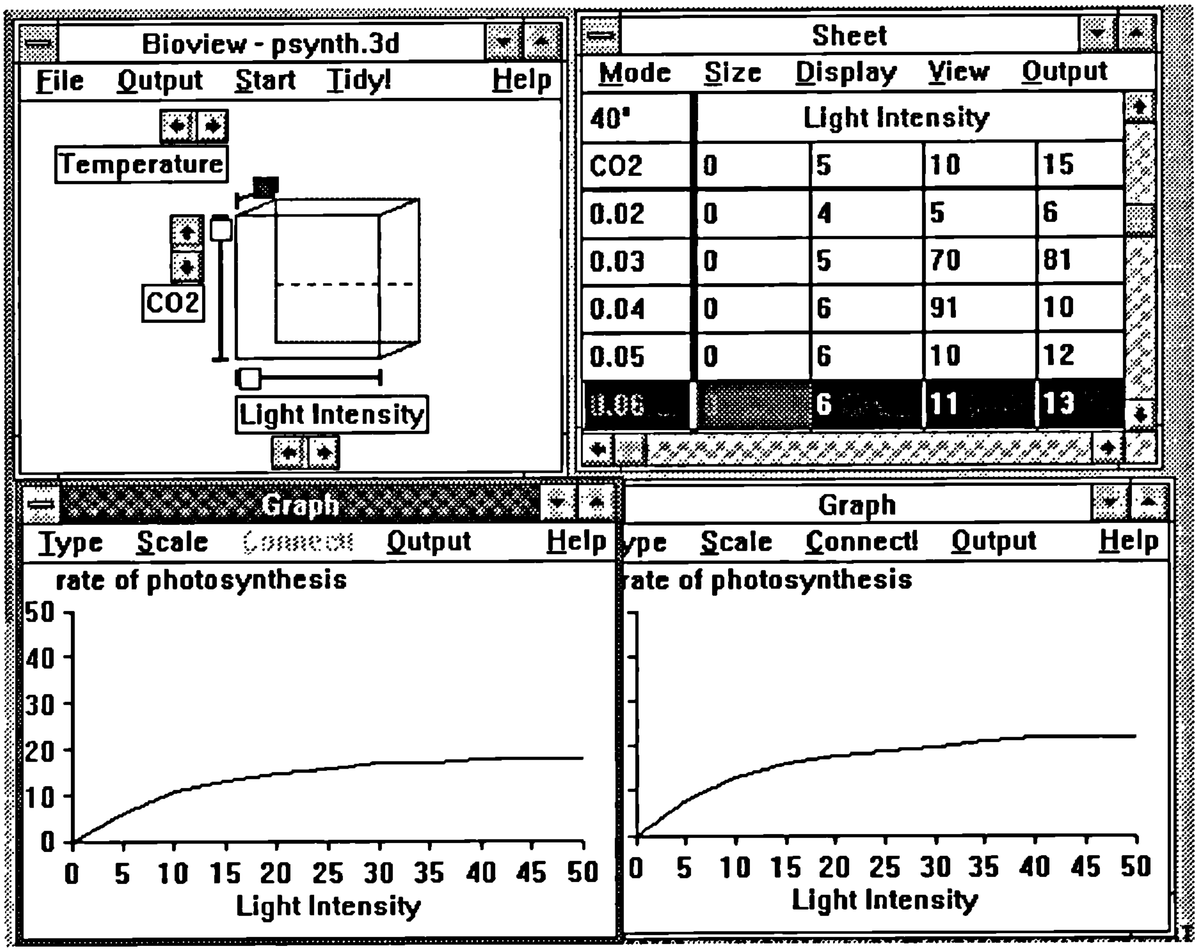


Figure A1.2: The effect of executing an m_sheet operation

The graph window in the bottom right hand corner of the screen shows the rate/L graph corresponding to the original T-sheet position, that is 25°C. Another graph window has been opened and connected as shown. The analysis window has been minimised and is now hidden as an icon at the bottom of the screen behind the left hand graph window. The graph shown in the new graph window corresponds to the new T-value, that is 40°C, and a section of the 40°C datasheet is now shown in the datasheet window.

In Figure A1.3 the result of executing an s_sheet(L) operation and an m_sheet(L) operation to locate the L-sheet at an intermediate position is illustrated. In addition s_col(T) and m_col(T) operations have been executed.

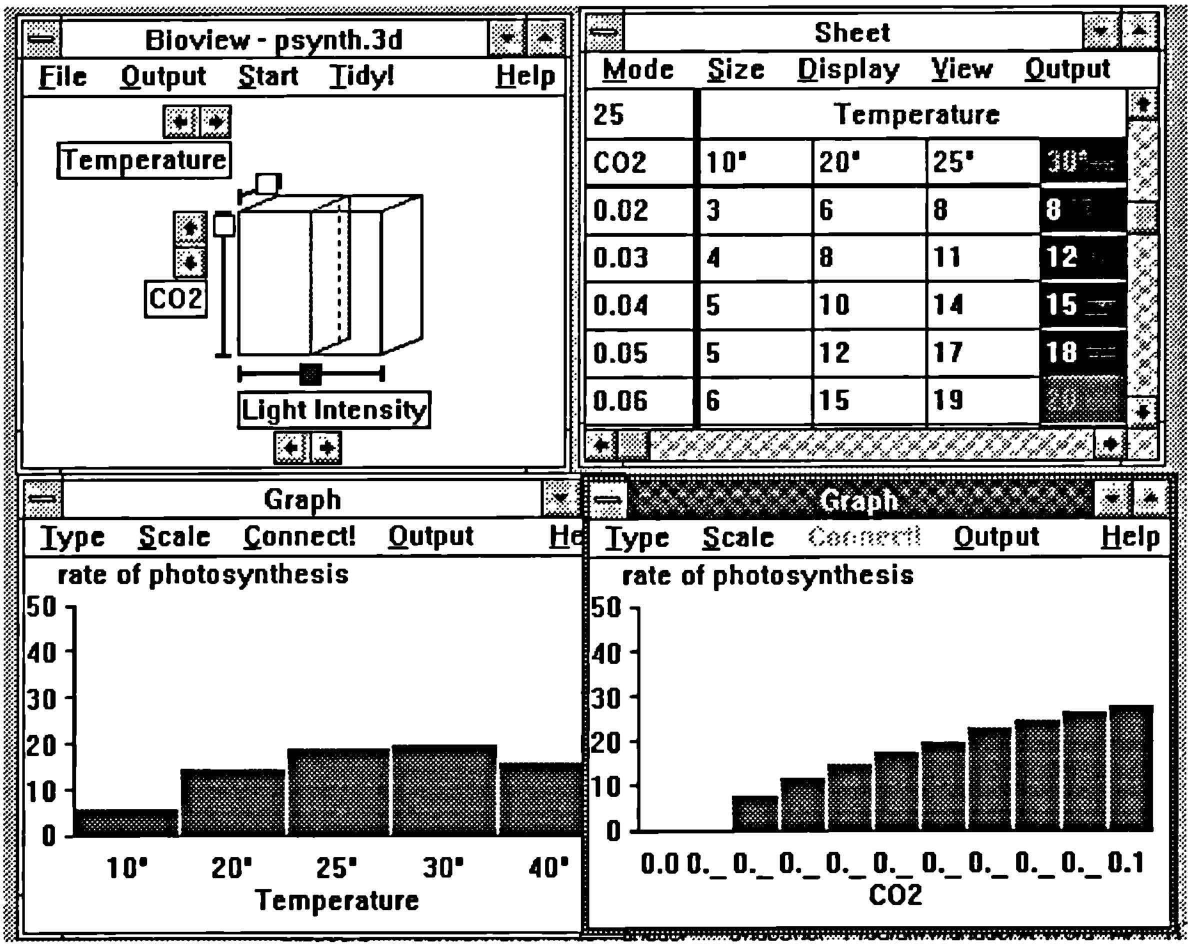


Figure A1.3: The effects of executing s_sheet(L), m_sheet(L), s_col(T), and m_col(T) operations

The rate/T graph shown in the unconnected left hand graph window corresponds to the effect of executing s_sheet(L) and m_sheet (L) operations. The rate/C graph shown in the connected right hand graph window shows the effect of executing an s_col(T) operation followed by an m_col(T) operation. The Type option has been used to select a bar graph in both of the displayed graph windows. Note that the C-values are shown in a truncated form due to the restricted size of the window.

Figures A1.2 and A1.3 illustrate how graphs can be compared by inspecting multiple graph windows. However, graphs can be "animated" by effecting rapid m_sheet or m_row/col operations by dragging a sheet slider or a row or column. Instances of a rate/variable graph are produced in rapid succession in this way, producing an animated effect.

Appendix 2

Successful method action strings

The action strings for successful methods are given below. Upper-case letters in brackets (an L, C, or T) designate the active system sub-register. The subscript (r or c) designates whether a row or column is being directly manipulated. Unit-task boundaries are marked by "||" symbols. Methods 4, 5, 6, 7, 8, and 9 consist of two methods executed in sequence chosen from Methods 1, 2, and 3. The action strings for each of the "double" methods are given for one order of this sequence. Action strings for the reverse order consist of the action strings for the component methods in reverse order.

Method 1

$1(L)_r$ || s_sheet(L) | m_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T)

$1(L)_c$ || s_sheet(L) | m_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C)

$1(C)_r$ || s_sheet(C) | m_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L)

$1(C)_c$ || s_sheet(C) | m_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T)

$1(T)_r$ || s_sheet(T) | m_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L)

$1(T)_c$ || s_sheet(T) | m_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C)

Method 2

$2(L)_r$ || s_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T)] || repeat [m_sheet(L) | inspect_graph(r/T)]

$2(L)_c$ || s_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C)] || repeat [m_sheet(L) | inspect_graph(r/C)]

$2(C)_r$ || s_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L)] || repeat [m_sheet(C) | inspect_graph(r/L)]

$2(C)_c$ || s_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T)] || repeat [m_sheet(C) | inspect_graph(r/T)]

$2(T)_r$ || s_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L)] || repeat [m_sheet(T) | inspect_graph(r/L)]

$2(T)_c$ || s_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C)] || repeat [m_sheet(T) | inspect_graph(r/C)]

Method 3

$3(L)_r$ || s_sheet(L) | m_sheet(L) || s_row(C) || inspect_graph(r/T)] || repeat [m_row(C) | inspect_graph(r/T)]

$3(L)_c$ || s_sheet(L) | m_sheet(L) || s_col(T) || inspect_graph(r/C)] || repeat [m_col (T) | inspect_graph(r/C)]

$3(C)_r$ || s_sheet(C) | m_sheet(C) || s_row(T) || inspect_graph(r/L)] || repeat [m_row(T) | inspect_graph(r/L)]

$3(C)_c$ || s_sheet(C) | m_sheet(C) || s_col(L) || inspect_graph(r/T)] || repeat [m_col (L) | inspect_graph(r/T)]

$3(T)_r$ || s_sheet(T) | m_sheet(T) || s_row(C) || inspect_graph(r/L)] || repeat [m_row(C) | inspect_graph(r/L)]

$3(T)_c$ || s_sheet(T) | m_sheet(T) || s_col(L) || inspect_graph(r/C)] || repeat [m_col (L) | inspect_graph(r/C)]

Method 4

$4(LL)_{rc}$ $1(L)_r$ & $1(L)_c$ || s_sheet(L) | m_sheet(L) || s_row(C) m_row(C) || inspect_graph(r/T)
|| s_col(T) | m_col(T) || inspect_graph(r/C)

$4(LC)_{rr}$ $1(L)_r$ & $1(C)_r$ || s_sheet(L) | m_sheet(L) || s_row(C) m_row(C) || inspect_graph(r/T)
|| s_sheet(C) | m_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L)

$4(LC)_{rc}$ $1(L)_r$ & $1(C)_c$ || s_sheet(L) | m_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T)
|| s_sheet(C) | m_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T)

$4(LT)_{rr}$ $1(L)_r$ & $1(T)_r$ || s_sheet(L) | m_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T)
|| s_sheet(T) | m_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L)

$4(LT)_{rc}$ $1(L)_r$ & $1(T)_c$ || s_sheet(L) | m_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T)
|| s_sheet(T) | m_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C)

$4(LC)_{cr}$ $1(L)_c$ & $1(C)_r$ || s_sheet(L) | m_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C)
|| s_sheet(C) | m_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L)

$4(LC)_{cc}$ $1(L)_c$ & $1(C)_c$ || s_sheet(L) | m_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C)
|| s_sheet(C) | m_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T)

$4(LT)_{cr}$ $1(L)_c$ & $1(T)_r$ || s_sheet(L) | m_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C)
|| s_sheet(T) | m_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L)

$4(LT)_{cc}$ $1(L)_c$ & $1(T)_c$ || s_sheet(L) | m_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C)
|| s_sheet(T) | m_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C)

$4(CC)_{rc}$ $1(C)_r$ & $1(C)_c$ || s_sheet(C) | m_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L)
|| s_col(L) | m_col(L) || inspect_graph(r/T)

$4(CT)_{rr}$ $1(C)_r$ & $1(T)_r$ || s_sheet(C) | m_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L)
|| s_sheet(T) | m_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L)

$4(CT)_{rc}$ $1(C)_r$ & $1(T)_c$ || s_sheet(C) | m_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L)
|| s_sheet(T) | m_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C)

$4(CT)_{cr}$ $1(C)_c$ & $1(T)_r$ s_sheet(C) | m_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T)
|| s_sheet(T) | m_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L)

$4(CT)_{cc}$ $1(C)_c$ & $1(T)_c$ || s_sheet(C) | m_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T)
|| s_sheet(T) | m_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C)

$4(TT)_{rc}$ $1(T)_r$ & $1(T)_c$ || s_sheet(T) | m_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L)
|| s_col(L) | m_col(L) || inspect_graph(r/C)

Method 5

$5(LL)_{rc}$ — $2(L)_r$ & $2(L)_c$ — || s_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T)] || repeat [m_sheet(L) | inspect_graph(r/T)] || s_col(T) | m_col(T) || inspect_graph(r/C) || repeat [m_sheet(L) | inspect_graph(r/C)]

$5(LC)_{rr}$ — $2(L)_r$ & $2(C)_r$ — || s_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T)] || repeat [m_sheet(L) | inspect_graph(r/T)] || s_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L) || repeat [m_sheet(C) | inspect_graph(r/L)]

$5(LC)_{rc}$ — $2(L)_r$ & $2(C)_c$ — || s_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T)] || repeat [m_sheet(L) | inspect_graph(r/T)] || s_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T) || repeat [m_sheet(C) | inspect_graph(r/T)]

$5(LT)_{rr}$ — $2(L)_r$ & $2(T)_r$ — || s_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T)] || repeat [m_sheet(L) | inspect_graph(r/T)] || s_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || repeat [m_sheet(T) | inspect_graph(r/L)]

$5(LT)_{rc}$ — $2(L)_r$ & $2(T)_c$ — || s_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T)] || repeat [m_sheet(L) | inspect_graph(r/T)] || s_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C) || repeat [m_sheet(T) | inspect_graph(r/C)]

$5(LC)_{cr}$ — $2(L)_c$ & $2(C)_r$ — || s_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C)] || repeat [m_sheet(L) | inspect_graph(r/C)] || s_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L) || repeat [m_sheet(C) | inspect_graph(r/L)]

$5(LC)_{cc}$ — $2(L)_c$ & $2(C)_c$ — || s_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C)] || repeat [m_sheet(L) | inspect_graph(r/C)] || s_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T)] || repeat [m_sheet(C) | inspect_graph(r/T)]

$5(LT)_{cr}$ — $2(L)_c$ & $2(T)_r$ — || s_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C)] || repeat [m_sheet(L) | inspect_graph(r/C)] || s_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || repeat [m_sheet(T) | inspect_graph(r/L)]

$5(LT)_{cc}$ — $2(L)_c$ & $2(T)_c$ — || s_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C)] || repeat [m_sheet(L) | inspect_graph(r/C)] || s_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C) || repeat [m_sheet(T) | inspect_graph(r/C)]

$5(CC)_{rc}$ — $2(C)_r$ & $2(C)_c$ — || s_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L)] || repeat [m_sheet(C) | inspect_graph(r/L)] || s_col(L) | m_col(L) || inspect_graph(r/T) || repeat [m_sheet(C) | inspect_graph(r/T)]

$5(CT)_{rr}$ — $2(C)_r$ & $2(T)_r$ — || s_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L)] || repeat [m_sheet(C) | inspect_graph(r/L)] || s_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || repeat [m_sheet(T) | inspect_graph(r/L)]

$5(CT)_{rc}$ — $2(C)_r$ & $2(T)_c$ — || s_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L)] || repeat [m_sheet(C) | inspect_graph(r/L)] || s_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C) || repeat [m_sheet(T) | inspect_graph(r/C)]

$5(CT)_{cr}$ $2(C)_c$ & $2(T)_r$ || s_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T)] || repeat [m_sheet(C) | inspect_graph(r/T)] || s_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || repeat [m_sheet(T) | inspect_graph(r/L)]

$5(CT)_{cc}$ $2(C)_c$ & $2(T)_c$ || s_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T)] || repeat [m_sheet(C) | inspect_graph(r/T)] || s_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C) || repeat [m_sheet(T) | inspect_graph(r/C)]

$5(TT)_{rc}$ $2(T)_r$ & $2(T)_c$ || s_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L)] || repeat [m_sheet(T) | inspect_graph(r/L)] || s_col(L) | m_col(L) || inspect_graph(r/C) || repeat [m_sheet(T) | inspect_graph(r/C)]

Method 6

$6(LL)_{rc}$ $3(L)_r$ & $3(L)_c$ || s_sheet(L) | m_sheet(L) || s_row(C) || inspect_graph(r/T)] || repeat [m_row(C) | inspect_graph(r/T)] || s_col(T) || inspect_graph(r/C)] || repeat [m_col (T) | inspect_graph(r/C)]

$6(LC)_{rr}$ $3(L)_r$ & $3(C)_r$ || s_sheet(L) | m_sheet(L) || s_row(C) || inspect_graph(r/T)] || repeat [m_row(C) | inspect_graph(r/T)] || s_sheet(C) | m_sheet(C) || s_row(T) || inspect_graph(r/L) || repeat [m_row(T) | inspect_graph(r/L)]

$6(LC)_{rc}$ $3(L)_r$ & $3(C)_c$ || s_sheet(L) | m_sheet(L) || s_row(C) || inspect_graph(r/T)] || repeat [m_row(C) | inspect_graph(r/T)] || s_sheet(C) | m_sheet(C) || s_col(L) || inspect_graph(r/T) || repeat [m_col (L) | inspect_graph(r/T)]

$6(LT)_{rr}$ $3(L)_r$ & $3(T)_r$ || s_sheet(L) | m_sheet(L) || s_row(C) || inspect_graph(r/T)] || repeat [m_row(C) | inspect_graph(r/T)] || s_sheet(T) | m_sheet(T) || s_row(C) || inspect_graph(r/L) || repeat [m_row(C) | inspect_graph(r/L)]

$6(LT)_{rc}$ $3(L)_r$ & $3(T)_c$ || s_sheet(L) | m_sheet(L) || s_row(C) || inspect_graph(r/T)] || repeat [m_row(C) | inspect_graph(r/T)] || s_sheet(T) | m_sheet(T) || s_col(L) || inspect_graph(r/C) || repeat [m_col (L) | inspect_graph(r/C)]

$6(LC)_{cr}$ $3(L)_c$ & $3(C)_r$ || s_sheet(L) | m_sheet(L) || s_col(T) || inspect_graph(r/C)] || repeat [m_col (T) | inspect_graph(r/C)] || s_sheet(C) | m_sheet(C) || s_row(T) || inspect_graph(r/L) || repeat [m_row(T) | inspect_graph(r/L)]

$6(LC)_{cc}$ $3(L)_c$ & $3(C)_c$ || s_sheet(L) | m_sheet(L) || s_col(T) || inspect_graph(r/C)] || repeat [m_col (T) | inspect_graph(r/C)] || s_sheet(C) | m_sheet(C) || s_col(L) ||inspect_graph(r/T) || repeat [m_col (L) | inspect_graph(r/T)]

$6(LT)_{cr}$ $3(L)_c$ & $3(T)_r$ || s_sheet(L) | m_sheet(L) || s_col(T) || inspect_graph(r/C)] || repeat [m_col (T) | inspect_graph(r/C)] || s_sheet(T) | m_sheet(T) || s_row(C) || inspect_graph(r/L) || repeat [m_row(C) | inspect_graph(r/L)]

$6(LT)_{cc}$ $3(L)_c$ & $3(T)_c$ || s_sheet(L) | m_sheet(L) || s_col(T) || inspect_graph(r/C)] || repeat [m_col (T)
| inspect_graph(r/C)] || s_sheet(T) | m_sheet(T) || s_col(L) ||inspect_graph(r/C)
|| repeat [m_col (L) | inspect_graph(r/C)]

$6(CC)_{rc}$ $3(C)_r$ & $3(C)_c$ || s_sheet(C) | m_sheet(C) || s_row(T) || inspect_graph(r/L)] || repeat [m_row(T)
| inspect_graph(r/L)] || s_col(L) || inspect_graph(r/T) || repeat [m_col (L)
| inspect_graph(r/T)]

$6(CT)_{rr}$ $3(C)_r$ & $3(T)_r$ || s_sheet(C) | m_sheet(C) || s_row(T) || inspect_graph(r/L)] || repeat [m_row(T)
| inspect_graph(r/L)] || s_sheet(T) | m_sheet(T) || s_row(C) || inspect_graph(r/L)
|| repeat [m_row(C) | inspect_graph(r/L)]

$6(CT)_{rc}$ $3(C)_r$ & $3(T)_c$ || s_sheet(C) | m_sheet(C) || s_row(T) || inspect_graph(r/L)] || repeat [m_row(T)
| inspect_graph(r/L)] || s_sheet(T) | m_sheet(T) || s_col(L) || inspect_graph(r/C)
|| repeat [m_col (L) | inspect_graph(r/C)]

$6(CT)_{cr}$ $3(C)_c$ & $3(T)_r$ || s_sheet(C) | m_sheet(C) || s_col(L) || inspect_graph(r/T)] || repeat [m_col (L)
| inspect_graph(r/T)] || s_sheet(T) | m_sheet(T) || s_row(C) || inspect_graph(r/L)
|| repeat [m_row(C) | inspect_graph(r/L)]

$6(CT)_{cc}$ $3(C)_c$ & $3(T)_c$ || s_sheet(C) | m_sheet(C) || s_col(L) || inspect_graph(r/T)] || repeat [m_col (L)
| inspect_graph(r/T)] || s_sheet(T) | m_sheet(T) || s_col(L) || inspect_graph(r/C)
|| repeat [m_col (L) | inspect_graph(r/C)]

$6(TT)_{rc}$ $3(T)_r$ & $3(T)_c$ || s_sheet(T) | m_sheet(T) || s_row(C) || inspect_graph(r/L)] || repeat [m_row(C)
| inspect_graph(r/L)] || s_col(L) || inspect_graph(r/C)] || repeat [m_col (L)
| inspect_graph(r/C)]

Method 7

$7(LL)_{rr}$ $1(L)_r$ & $2(L)_r$ || s_sheet(L) | m_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T)
|| s_row(C) | m_row(C) || inspect_graph(r/T) || repeat [m_sheet(L)
| inspect_graph(r/T)]

$7(LL)_{rc}$ $1(L)_r$ & $2(L)_c$ || s_sheet(L) | m_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T)
|| s_col(T) | m_col(T) || inspect_graph(r/C) || repeat [m_sheet(L)
| inspect_graph(r/C)]

$7(LC)_{rr}$ $1(L)_r$ & $2(C)_r$ || s_sheet(L) | m_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T)
|| s_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L) || repeat [m_sheet(C)
| inspect_graph(r/L)]

$7(LC)_{rc}$ $1(L)_r$ & $2(C)_c$ || s_sheet(L) | m_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T)
|| s_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T) || repeat [m_sheet(C)
| inspect_graph(r/T)]

$7(LT)_{rr}$ $1(L)_r$ & $2(T)_r$ || s_sheet(L) | m_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T)
|| s_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || repeat [m_sheet(T)
| inspect_graph(r/L)]

$7(LT)_{rc}$ $1(L)_r$ & $2(T)_c$ || s_sheet(L) | m_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T)
|| s_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C) || repeat [m_sheet(T)
| inspect_graph(r/C)]

$7(LL)_{cr}$ $1(L)_c$ & $2(L)_r$ || s_sheet(L) | m_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C)
|| s_row(C) | m_row(C) || inspect_graph(r/T)] || repeat [m_sheet(L)
| inspect_graph(r/T)]

$7(LL)_{cc}$ $1(L)_c$ & $2(L)_c$ || s_sheet(L) | m_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C)
|| s_col(T) | m_col(T) || inspect_graph(r/C) || repeat [m_sheet(L)
| inspect_graph(r/C)]

$7(LC)_{cr}$ $1(L)_c$ & $2(C)_r$ || s_sheet(L) | m_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C)
|| s_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L) || repeat [m_sheet(C)
| inspect_graph(r/L)]

$7(LC)_{cc}$ $1(L)_c$ & $2(C)_c$ || s_sheet(L) | m_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C)
|| s_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T) || repeat [m_sheet(C)
| inspect_graph(r/T)]

$7(LT)_{cr}$ $1(L)_c$ & $2(T)_r$ || s_sheet(L) | m_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C)
|| s_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || repeat [m_sheet(T)
| inspect_graph(r/L)]

$7(LT)_{cc}$ $1(L)_c$ & $2(T)_c$ || s_sheet(L) | m_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C)
|| s_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C) || repeat [m_sheet(T)
| inspect_graph(r/C)]

$7(CL)_{rr}$ $1(C)_r$ & $2(L)_r$ || s_sheet(C) | m_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L)
|| s_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T) || repeat [m_sheet(L)
| inspect_graph(r/T)]

$7(CL)_{rc}$ $1(C)_r$ & $2(L)_c$ || s_sheet(C) | m_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L)
|| s_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C) || repeat [m_sheet(L)
| inspect_graph(r/C)]

$7(CC)_{rr}$ $1(C)_r$ & $2(C)_r$ || s_sheet(C) | m_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L)
|| s_row(T) | m_row(T) || inspect_graph(r/L) || repeat [m_sheet(C)
| inspect_graph(r/L)]

$7(CC)_{rc}$ $1(C)_r$ & $2(C)_c$ || s_sheet(C) | m_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L)
|| s_col(L) | m_col(L) || inspect_graph(r/T) || repeat [m_sheet(C)
| inspect_graph(r/T)]

$7(CT)_{rr}$ $1(C)_r$ & $2(T)_r$ || s_sheet(C) | m_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L) || s_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || repeat [m_sheet(T) | inspect_graph(r/L)]

$7(CT)_{rc}$ $1(C)_r$ & $2(T)_c$ || s_sheet(C) | m_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L) || s_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C) || repeat [m_sheet(T) | inspect_graph(r/C)]

$7(CL)_{cr}$ $1(C)_c$ & $2(L)_r$ || s_sheet(C) | m_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T) || s_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T) || repeat [m_sheet(L) | inspect_graph(r/T)]

$7(CL)_{cc}$ $1(C)_c$ & $2(L)_c$ || s_sheet(C) | m_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T) || s_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T) || repeat [m_sheet(L) | inspect_graph(r/T)]

$7(CC)_{cr}$ $1(C)_c$ & $2(C)_r$ || s_sheet(C) | m_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T) || s_row(T) | m_row(T) || inspect_graph(r/L) || repeat [m_sheet(C) | inspect_graph(r/L)]

$7(CC)_{cc}$ $1(C)_c$ & $2(C)_c$ || s_sheet(C) | m_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T) || s_col(L) | m_col(L) || inspect_graph(r/T) || repeat [m_sheet(C) | inspect_graph(r/T)]

$7(CT)_{cr}$ $1(C)_c$ & $2(T)_r$ || s_sheet(C) | m_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T) || s_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || repeat [m_sheet(T) | inspect_graph(r/L)]

$7(CT)_{cc}$ $1(C)_c$ & $2(T)_c$ || s_sheet(C) | m_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T) || s_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C) || repeat [m_sheet(T) | inspect_graph(r/C)]

$7(TL)_{rr}$ $1(T)_r$ & $2(L)_r$ || s_sheet(T) | m_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || s_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T) || repeat [m_sheet(L) | inspect_graph(r/T)]

$7(TL)_{rc}$ $1(T)_r$ & $2(L)_c$ || s_sheet(T) | m_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || s_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/T) || repeat [m_sheet(L) | inspect_graph(r/T)]

$7(TC)_{rr}$ $1(T)_r$ & $2(C)_r$ || s_sheet(T) | m_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || s_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L) || repeat [m_sheet(C) | inspect_graph(r/L)]

$7(TC)_{rc}$ $1(T)_r$ & $2(C)_c$ || s_sheet(T) | m_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || s_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T) || repeat [m_sheet(C) | inspect_graph(r/T)]

$7(TT)_{rr}$ $1(T)_r$ & $2(T)_r$ || s_sheet(T) | m_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L)
|| s_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L) || repeat [m_sheet(C)
| inspect_graph(r/L)]

$7(TT)_{rc}$ $1(T)_r$ & $2(T)_c$ || s_sheet(T) | m_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L)
|| s_col(L) | m_col(L) || inspect_graph(r/C) || repeat [m_sheet(T)
| inspect_graph(r/C)]

$7(TL)_{cr}$ $1(T)_c$ & $2(L)_r$ || s_sheet(T) | m_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C)
|| s_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T) || repeat [m_sheet(L)
| inspect_graph(r/T)]

$7(TL)_{cc}$ $1(T)_c$ & $2(L)_c$ || s_sheet(T) | m_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C)
|| s_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/T) || repeat [m_sheet(L)
| inspect_graph(r/C)]

$7(TC)_{cr}$ $1(T)_c$ & $2(C)_r$ || s_sheet(T) | m_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C)
|| s_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L) || repeat [m_sheet(C)
| inspect_graph(r/L)]

$7(TC)_{cc}$ $1(T)_c$ & $2(C)_c$ || s_sheet(T) | m_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C)
|| s_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T) || repeat [m_sheet(C)
| inspect_graph(r/T)]

$7(TT)_{cr}$ $1(T)_c$ & $2(T)_r$ || s_sheet(T) | m_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C)
|| s_row(C) | m_row(C) || inspect_graph(r/L) || repeat [m_sheet(T)
| inspect_graph(r/L)]

$7(TT)_{cc}$ $1(T)_c$ & $2(T)_c$ || s_sheet(T) | m_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C)
|| s_col(L) | m_col(L) || inspect_graph(r/C) || repeat [m_sheet(T)
| inspect_graph(r/C)

Method 8

$8(LL)_{rr}$ $1(L)_r$ & $3(L)_r$ || s_sheet(L) | m_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T)
|| s_row(C) || inspect_graph(r/T)] || repeat [m_row(C) | inspect_graph(r/L)]

$8(LL)_{rc}$ $1(L)_r$ & $3(L)_c$ || s_sheet(L) | m_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T)
|| s_col(T) || inspect_graph(r/C)] || repeat [m_col(T) | inspect_graph(r/C)]

$8(LC)_{rr}$ $1(L)_r$ & $3(C)_r$ || s_sheet(L) | m_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T)
|| s_sheet(C) | m_sheet(C) || s_row(T) || inspect_graph(r/L)] || repeat [m_row(T)
| inspect_graph(r/L)]

$8(LC)_{rc}$ $1(L)_r$ & $3(C)_c$ || s_sheet(L) | m_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T)
|| s_sheet(C) | m_sheet(C) || s_col(L) || inspect_graph(r/T)] || repeat [m_col (L)
| inspect_graph(r/T)]

$8(LT)_{rr}$ $1(L)_r$ & $3(T)_r$ || s_sheet(L) | m_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T) || s_sheet(T) | m_sheet(T) || s_row(C) || inspect_graph(r/L)] || repeat [m_row(C) | inspect_graph(r/L)]

$8(LT)_{rc}$ $1(L)_r$ & $3(T)_c$ || s_sheet(L) | m_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T) || s_sheet(T) | m_sheet(T) || s_col(L) || inspect_graph(r/C)] || repeat [m_col (L) | inspect_graph(r/C)]

$8(LL)_{cr}$ $1(L)_c$ & $3(L)_r$ || s_sheet(L) | m_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C) || s_row(C) || inspect_graph(r/T)] || repeat [m_row(C) | inspect_graph(r/T)]

$8(LL)_{cc}$ $1(L)_c$ & $3(L)_c$ || s_sheet(L) | m_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C) || s_row(C) || inspect_graph(r/T)] || repeat [m_row(C) | inspect_graph(r/T)]

$8(LC)_{cr}$ $1(L)_c$ & $3(C)_r$ || s_sheet(L) | m_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C) || s_sheet(C) | m_sheet(C) || s_row(T) || inspect_graph(r/L)] || repeat [m_row(T) | inspect_graph(r/L)]

$8(LC)_{cc}$ $1(L)_c$ & $3(C)_c$ || s_sheet(L) | m_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C) || s_sheet(C) | m_sheet(C) || s_col(L) || inspect_graph(r/T)] || repeat [m_col (L) | inspect_graph(r/T)]

$8(LT)_{cr}$ $1(L)_c$ & $3(T)_r$ || s_sheet(L) | m_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C) || s_sheet(T) | m_sheet(T) || s_row(C) || inspect_graph(r/L)] || repeat [m_row(C) | inspect_graph(r/L)]

$8(LT)_{cc}$ $1(L)_c$ & $3(T)_c$ || s_sheet(L) | m_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C) || s_sheet(T) | m_sheet(T) || s_col(L) || inspect_graph(r/C)] || repeat [m_col (L) | inspect_graph(r/C)]

$8(CL)_{rr}$ $1(C)_r$ & $3(L)_r$ || s_sheet(C) | m_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L) || s_sheet(L) | m_sheet(L) || s_row(C) || inspect_graph(r/T)] || repeat [m_row(C) | inspect_graph(r/T)]

$8(CL)_{rc}$ $1(C)_r$ & $3(L)_c$ || s_sheet(C) | m_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L) || s_sheet(L) | m_sheet(L) || s_row(C) || inspect_graph(r/T)] || repeat [m_row(C) | inspect_graph(r/T)]

$8(CC)_{rr}$ $1(C)_r$ & $3(C)_r$ || s_sheet(C) | m_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L) || s_row(T) || inspect_graph(r/L)] || repeat [m_row(T) | inspect_graph(r/L)]

$8(CC)_{rc}$ $1(C)_r$ & $3(C)_c$ || s_sheet(C) | m_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L) || s_col(L) || inspect_graph(r/T)] || repeat [m_col (L) | inspect_graph(r/T)]

$8(CT)_{rr}$ $1(C)_r$ & $3(T)_r$ || s_sheet(C) | m_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L) || s_sheet(T) | m_sheet(T) || s_row(C) || inspect_graph(r/L)] || repeat [m_row(C) | inspect_graph(r/L)]

$8(CT)_{rc}$ $1(C)_r$ & $3(T)_c$ || s_sheet(C) | m_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L) || s_sheet(T) | m_sheet(T) || s_col(L) || inspect_graph(r/C)] || repeat [m_col (L) | inspect_graph(r/C)]

$8(CL)_{cr}$ $1(C)_{c}$ & $3(L)_{r}$ || s_sheet(C) | m_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T) || s_sheet(L) | m_sheet(L) || s_row(C) || inspect_graph(r/T)] || repeat [m_row(C) | inspect_graph(r/T)]

$8(CL)_{cc}$ $1(C)_{c}$ & $3(L)_{c}$ || s_sheet(C) | m_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T) || s_sheet(L) | m_sheet(L) || s_col(T) || inspect_graph(r/C)] || repeat [m_col(T) | inspect_graph(r/T)]

$8(CC)_{cr}$ $1(C)_{c}$ & $3(C)_{r}$ || s_sheet(C) | m_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T) || s_row(T) || inspect_graph(r/L)] || repeat [m_row(T) | inspect_graph(r/L)]

$8(CC)_{cc}$ $1(C)_{c}$ & $3(C)_{c}$ || s_sheet(C) | m_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T) || s_col(L) || inspect_graph(r/T)] || repeat [m_col (L) | inspect_graph(r/T)]

$8(CT)_{cr}$ $1(C)_{c}$ & $3(T)_{r}$ || s_sheet(C) | m_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T) || s_sheet(T) | m_sheet(T) || s_row(C) || inspect_graph(r/L)] || repeat [m_row(C) | inspect_graph(r/L)]

$8(CT)_{cc}$ $1(C)_{c}$ & $3(T)_{c}$ || s_sheet(C) | m_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T) || s_sheet(T) | m_sheet(T) || s_col(L) || inspect_graph(r/C)] || repeat [m_col (L) | inspect_graph(r/C)]

$8(TL)_{rr}$ $1(T)_{r}$ & $3(L)_{r}$ || s_sheet(T) | m_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || s_sheet(L) | m_sheet(L) || s_row(C) || inspect_graph(r/T)] || repeat [m_row(C) | inspect_graph(r/T)]

$8(TL)_{rc}$ $1(T)_{r}$ & $3(L)_{c}$ || s_sheet(T) | m_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || s_sheet(L) | m_sheet(L) || s_col(T) || inspect_graph(r/C)] || repeat [m_row(T) | inspect_graph(r/C)]

$8(TC)_{rr}$ $1(T)_{r}$ & $3(C)_{r}$ || s_sheet(T) | m_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || s_sheet(C) | m_sheet(C) || s_row(T) || inspect_graph(r/L)] || repeat [m_row(T) | inspect_graph(r/L)]

$8(TC)_{rc}$ $1(T)_{r}$ & $3(C)_{c}$ || s_sheet(T) | m_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || s_sheet(C) | m_sheet(C) || s_col(L) || inspect_graph(r/T)] || repeat [m_col (L) | inspect_graph(r/T)]

$8(TT)_{rr}$ $1(T)_{r}$ & $3(T)_{r}$ || s_sheet(T) | m_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || s_row(C) || inspect_graph(r/L)] || repeat [m_row(C) | inspect_graph(r/L)]

$8(TT)_{rc}$ $1(T)_{r}$ & $3(T)_{c}$ || s_sheet(T) | m_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || s_col(L) || inspect_graph(r/C)] || repeat [m_col (L) | inspect_graph(r/C)]

$8(TL)_{cr}$ $1(T)_{c}$ & $3(L)_{r}$ || s_sheet(T) | m_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C) || s_sheet(L) | m_sheet(L) || s_row(C) || inspect_graph(r/T)] || repeat [m_row(C) | inspect_graph(r/T)]

$8(TL)_{cc}$ $1(T)_{c}$ & $3(L)_{c}$ || s_sheet(T) | m_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C) || s_sheet(L) | m_sheet(L) || s_col(T) || inspect_graph(r/C)] || repeat [m_col(T) | inspect_graph(r/C)]

$8(TC)_{cr}$ $1(T)_c$ & $3(C)_r$ || s_sheet(T) | m_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C)
|| s_sheet(C) | m_sheet(C) || s_row(T) || inspect_graph(r/L)] || repeat [m_row(T)
| inspect_graph(r/L)]

$8(TC)_{cc}$ $1(T)_c$ & $3(C)_c$ || s_sheet(T) | m_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C)
|| s_sheet(C) | m_sheet(C) || s_col(L) || inspect_graph(r/T)] || repeat [m_col (L)
| inspect_graph(r/T)]

$8(TT)_{cr}$ $1(T)_c$ & $3(T)_r$ || s_sheet(T) | m_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C)
|| s_row(C) || inspect_graph(r/L)] || repeat [m_row(C) | inspect_graph(r/L)]

$8(TT)_{cc}$ $1(T)_c$ & $3(T)_c$ || s_sheet(T) | m_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C)
|| s_col(L) || inspect_graph(r/C)] || repeat [m_col (L) | inspect_graph(r/C)]

Method 9

$9(LL)_{rr}$ $2(L)_r$ & $3(L)_r$ || s_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T) || repeat [m_sheet(L)
| inspect_graph(r/T)] || m_sheet(L) || s_row(C) ||inspect_graph(r/T)
|| repeat [m_row(C) | inspect_graph(r/T)]

$9(LL)_{rc}$ $2(L)_r$ & $3(L)_c$ || s_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T) || repeat [m_sheet(L)
| inspect_graph(r/T)] || m_sheet(L) || s_col(T) ||inspect_graph(r/C)
|| repeat [m_col(T) | inspect_graph(r/C)]

$9(LC)_{rr}$ $2(L)_r$ & $3(C)_r$ || s_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T) || repeat [m_sheet(L)
| inspect_graph(r/T) || s_sheet(C) | m_sheet(C) || s_row(T)
|| inspect_graph(r/L)] || repeat [m_row(T) | inspect_graph(r/L)]

$9(LC)_{rc}$ $2(L)_r$ & $3(C)_c$ || s_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T) || repeat [m_sheet(L)
| inspect_graph(r/T) || s_sheet(C) | m_sheet(C) || s_col(L) || inspect_graph(r/T)]
|| repeat [m_col (L) | inspect_graph(r/T)]

$9(LT)_{rr}$ $2(L)_r$ & $3(T)_r$ || s_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T) || repeat [m_sheet(L)
| inspect_graph(r/T) || s_sheet(T) | m_sheet(T) || s_row(C)
|| inspect_graph(r/L)] || repeat [m_row(C) | inspect_graph(r/L)]

$9(LT)_{rc}$ $2(L)_r$ & $3(T)_c$ || s_sheet(L) || s_row(C) | m_row(C) || inspect_graph(r/T) || repeat [m_sheet(L)
| inspect_graph(r/T)] || s_sheet(T) | m_sheet(T) || s_col(L) || inspect_graph(r/C)
|| repeat [m_col (L) | inspect_graph(r/C)] ||

$9(LL)_{cr}$ $2(L)_c$ & $3(L)_r$ || s_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C) || repeat [m_sheet(L)
| inspect_graph(r/C)] || m_sheet(L) || s_row(C) || inspect_graph(r/T)
|| repeat [m_row(C) | inspect_graph(r/T)]

$9(LL)_{cc}$ $2(L)_c$ & $3(L)_c$ || s_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C) || repeat [m_sheet(L)
| inspect_graph(r/C)] || m_sheet(L) || s_col(T) || inspect_graph(r/C)
|| repeat [m_col(T) | inspect_graph(r/C)]

<table>
<tr><td>$9(LC)_{cr}$</td><td>$2(L)_c$ & $3(C)_r$</td><td>|| s_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C) || repeat [m_sheet(L)
| inspect_graph(r/C) || s_sheet(C) | m_sheet(C) || s_row(T) || inspect_graph(r/L)] ||
repeat [m_row(T) | inspect_graph(r/L)]</td></tr>
<tr><td>$9(LC)_{cc}$</td><td>$2(L)_c$ & $3(C)_c$</td><td>|| s_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C) || repeat [m_sheet(L)
| inspect_graph(r/C)] || s_sheet(C) | m_sheet(C) || s_col(L) || inspect_graph(r/T)
|| repeat [m_col (L) | inspect_graph(r/T)]</td></tr>
<tr><td>$9(LT)_{cr}$</td><td>$2(L)_c$ & $3(T)_r$</td><td>|| s_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C) || repeat [m_sheet(L)
| inspect_graph(r/C) || s_sheet(T) | m_sheet(T) || s_row(C) || inspect_graph(r/L)]
|| repeat [m_row(C) | inspect_graph(r/L)]</td></tr>
<tr><td>$9(LT)_{cc}$</td><td>$2(L)_c$ & $3(T)_c$</td><td>|| s_sheet(L) || s_col(T) | m_col(T) || inspect_graph(r/C) || repeat [m_sheet(L)
| inspect_graph(r/C)] || s_sheet(T) | m_sheet(T) || s_col(L) || inspect_graph(r/C)
|| repeat [m_col (L) | inspect_graph(r/C)]</td></tr>
<tr><td>$9(CL)_{rr}$</td><td>$2(C)_r$ & $3(L)_r$</td><td>|| s_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L) || repeat [m_sheet(C)
| inspect_graph(r/L)] || s_sheet(L) | m_sheet(L) || s_row(C) || inspect_graph(r/T)
|| repeat [m_row(C) | inspect_graph(r/T)]</td></tr>
<tr><td>$9(CL)_{rc}$</td><td>$2(C)_r$ & $3(L)_c$</td><td>|| s_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L) || repeat [m_sheet(C)
| inspect_graph(r/L)] || s_sheet(L) | m_sheet(L) || s_col(T) || inspect_graph(r/C) ||
repeat [m_col(T) | inspect_graph(r/C)]</td></tr>
<tr><td>$9(CC)_{rr}$</td><td>$2(C)_r$ & $3(C)_r$</td><td>|| s_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L) || repeat [m_sheet(C)
| inspect_graph(r/L)] || m_sheet(C) || s_row(T) || inspect_graph(r/L)
|| repeat [m_row(T) | inspect_graph(r/L)]</td></tr>
<tr><td>$9(CC)_{rc}$</td><td>$2(C)_r$ & $3(C)_c$</td><td>|| s_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L) || repeat [m_sheet(C)
| inspect_graph(r/L) || m_sheet(C) || s_col(L) || inspect_graph(r/T)]
|| repeat [m_col (L) | inspect_graph(r/T)]</td></tr>
<tr><td>$9(CT)_{rr}$</td><td>$2(C)_r$ & $3(T)_r$</td><td>|| s_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L) || repeat [m_sheet(C)
| inspect_graph(r/L)] || s_sheet(T) | m_sheet(T) || s_row(C) || inspect_graph(r/L)
|| repeat [m_row(C) | inspect_graph(r/L)]</td></tr>
<tr><td>$9(CT)_{rc}$</td><td>$2(C)_r$ & $3(T)_c$</td><td>|| s_sheet(C) || s_row(T) | m_row(T) || inspect_graph(r/L) || repeat [m_sheet(C)
| inspect_graph(r/L)] || s_sheet(T) | m_sheet(T) || s_col(L) || inspect_graph(r/C)
|| repeat [m_col (L) | inspect_graph(r/C)]</td></tr>
<tr><td>$9(CL)_{cr}$</td><td>$2(C)_c$ & $3(L)_r$</td><td>|| s_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T) || repeat [m_sheet(C)
| inspect_graph(r/T)] || s_sheet(L) | m_sheet(L) || s_row(C) || inspect_graph(r/T)
|| repeat [m_row(C) | inspect_graph(r/T)]</td></tr>
<tr><td>$9(CL)_{cc}$</td><td>$2(C)_c$ & $3(L)_c$</td><td>|| s_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T) || repeat [m_sheet(C)
| inspect_graph(r/T)] || s_sheet(L) | m_sheet(L) || s_col(C) || inspect_graph(r/C)
|| repeat [m_col(T) | inspect_graph(r/C)]</td></tr>
</table>

$9(CC)_{cr}$ $2(C)_c$ & $3(C)_r$ || s_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T)] || repeat [m_sheet(C) | inspect_graph(r/T) || m_sheet(C) || s_row(T) || inspect_graph(r/L) || repeat [m_row(T) | inspect_graph(r/L)]

$9(CC)_{cc}$ $2(C)_c$ & $3(C)_c$ || s_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T) || repeat [m_sheet(C) | inspect_graph(r/T)] || m_sheet(C) || s_col(L) || inspect_graph(r/T) || repeat [m_col (L) | inspect_graph(r/T)]

$9(CT)_{cr}$ $2(C)_c$ & $3(T)_r$ || s_sheet(C) || s_col(L) m_col(L) || inspect_graph(r/T) || repeat [m_sheet(C) | inspect_graph(r/T)] || s_sheet(T) | m_sheet(T) || s_row(C) || inspect_graph(r/L) || repeat [m_row(C) | inspect_graph(r/L)]

$9(CT)_{cc}$ $2(C)_c$ & $3(T)_c$ || s_sheet(C) || s_col(L) | m_col(L) || inspect_graph(r/T) || repeat [m_sheet(C) | inspect_graph(r/T)] || s_sheet(T) | m_sheet(T) || s_col(L) || inspect_graph(r/C) || repeat [m_col (L) | inspect_graph(r/C)]

$9(TL)_{rr}$ $2(T)_r$ & $3(L)_r$ || s_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || repeat [m_sheet(T) | inspect_graph(r/L)] || s_sheet(L) | m_sheet(L) || s_row(C) || inspect_graph(r/T) || repeat [m_row(C) | inspect_graph(r/T)]

$9(TL)_{rc}$ $2(T)_r$ & $3(L)_c$ || s_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || repeat [m_sheet(T) | inspect_graph(r/L)] || s_sheet(L) | m_sheet(L) || s_col(T) || inspect_graph(r/C) || repeat [m_col(T) | inspect_graph(r/C)]

$9(TC)_{rr}$ $2(T)_r$ & $3(C)_r$ || s_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || repeat [m_sheet(T) | inspect_graph(r/L)] || s_sheet(C) | m_sheet(C) || s_row(T) || inspect_graph(r/L) || repeat [m_row(T) | inspect_graph(r/L)]

$9(TC)_{rc}$ $2(T)_r$ & $3(C)_c$ || s_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || repeat [m_sheet(T) | inspect_graph(r/L)] || s_sheet(C) | m_sheet(C) || s_col(L) || inspect_graph(r/T) || repeat [m_col (L) | inspect_graph(r/T)]

$9(TT)_{rr}$ $2(T)_r$ & $3(T)_r$ || s_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || repeat [m_sheet(T) | inspect_graph(r/L)] || m_sheet(T) || s_row(C) || inspect_graph(r/T) || repeat [m_row(T) | inspect_graph(r/T)]

$9(TT)_{rc}$ $2(T)_r$ & $3(T)_c$ || s_sheet(T) || s_row(C) | m_row(C) || inspect_graph(r/L) || repeat [m_sheet(T) | inspect_graph(r/L)] || m_sheet(T) || s_col(L) || inspect_graph(r/C) || repeat [m_col (L) | inspect_graph(r/C)]

$9(TL)_{cr}$ $2(T)_c$ & $3(L)_r$ || s_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C) || repeat [m_sheet(T) | inspect_graph(r/C)] || s_sheet(L) | m_sheet(L) || s_row(C) || inspect_graph(r/T) || repeat [m_row(C) | inspect_graph(r/T)]

$9(TL)_{cc}$ $2(T)_c$ & $3(L)_c$ || s_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C) || repeat [m_sheet(T) | inspect_graph(r/C) || s_sheet(L) | m_sheet(L) || s_col(T) || inspect_graph(r/C)] || repeat [m_col(T) | inspect_graph(r/C)]

$9(TC)_{cr}$ $2(T)_c$ & $3(C)_r$ || s_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C) || repeat [m_sheet(T)
| inspect_graph(r/C)] || s_sheet(C) | m_sheet(C) || s_row(T) || inspect_graph(r/L)
|| repeat [m_row(T) | inspect_graph(r/L)]

$9(TC)_{cc}$ $2(T)_c$ & $3(C)_c$ || s_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C) || repeat [m_sheet(T)
| inspect_graph(r/C)] || s_sheet(C) | m_sheet(C) || s_col(L) || inspect_graph(r/T)
|| repeat [m_col (L) | inspect_graph(r/T)]

$9(TT)_{cr}$ $2(T)_c$ & $3(T)_r$ || s_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C) || repeat [m_sheet(T)
| inspect_graph(r/C)] || m_sheet(T) || s_row(C) || inspect_graph(r/L)
|| repeat [m_row(C) | inspect_graph(r/L)]

$9(TT)_{cc}$ $2(T)_c$ & $3(T)_c$ || s_sheet(T) || s_col(L) | m_col(L) || inspect_graph(r/C) || repeat [m_sheet(T)
| inspect_graph(r/C)] || m_sheet(T) || s_col(L) || inspect_graph(r/C)
|| repeat [m_col (L) | inspect_graph(r/C)]

Appendix 3

Action string length look-up tables

<table>
<tr><th>Truncation</th><th>Unit-task 1</th><th>Unit-task 2</th><th>Unit-task 3</th><th>Unit-task 4</th><th>n_e-n_t</th></tr>
<tr><td></td><td colspan="4"><--Method 1--></td><td></td></tr>
<tr><td>None</td><td>|| s_sheet | m_sheet</td><td>|| s_row/col | m_row/col</td><td>|| inspect_graph</td><td>-</td><td></td></tr>
<tr><td>Display (s_sheet)</td><td>|| m_sheet</td><td>|| s_row/col | m_row/col</td><td>|| inspect_graph</td><td>-</td><td>1</td></tr>
<tr><td>Display (m_sheet)</td><td>|| s_sheet</td><td>|| s_row/col | m_row/col</td><td>|| inspect_graph</td><td>-</td><td>1</td></tr>
<tr><td>Display (s-row/col)</td><td>|| s_sheet | m_sheet</td><td>|| m_row/col</td><td>|| inspect_graph</td><td>-</td><td>1</td></tr>
<tr><td>Display (m_row/col)</td><td>|| s_sheet | m_sheet</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>-</td><td>1</td></tr>
<tr><td></td><td colspan="4"><--Method 2--></td><td></td></tr>
<tr><td>None</td><td>|| s_sheet</td><td>|| s_row/col | m_row/col</td><td>|| inspect_graph</td><td>|| r[m_sheet | inspect_graph]</td><td></td></tr>
<tr><td>Display (s_sheet)</td><td>-</td><td>|| s_row | m_row/col</td><td>|| inspect_graph</td><td>|| r[m_sheet | inspect_graph]</td><td>1</td></tr>
<tr><td>Display (s_row/col)</td><td>|| s_sheet</td><td>|| m_row/col</td><td>|| inspect_graph</td><td>|| r[m_sheet | inspect_graph]</td><td>1</td></tr>
<tr><td>Display (m_row/col)</td><td>|| s_sheet</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>|| r[m_sheet | inspect_graph]</td><td>1</td></tr>
<tr><td>Task (inspect_graph)</td><td>|| s_sheet</td><td>|| s_row/col | m_row/col</td><td>-</td><td>|| r[m_sheet | inspect_graph]</td><td>1</td></tr>
<tr><td></td><td colspan="4"><--Method 3--></td><td></td></tr>
<tr><td>None</td><td>|| s_sheet | m_sheet</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>|| r[m_row/col | inspect_graph]</td><td></td></tr>
<tr><td>Display (s_sheet)</td><td>|| m_sheet</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>|| r[m_row/col | inspect_graph]</td><td>1</td></tr>
<tr><td>Display (m_sheet)</td><td>|| s_sheet</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>|| r[m_row/col | inspect_graph]</td><td>1</td></tr>
<tr><td>Display (s_row/col)</td><td>|| s_sheet | m_sheet</td><td>-</td><td>|| inspect_graph</td><td>|| r[m_row/col | inspect_graph]</td><td>1</td></tr>
<tr><td>Task (inspect_graph)</td><td>|| s_sheet | m_sheet</td><td>|| s_row/col</td><td>-</td><td>|| r[m_row/col | inspect_graph]</td><td>1</td></tr>
</table>

Truncation	Unit-task 1	Unit-task 2	Unit-task 3	Unit-task 4	n_e-n_t
	<------------	------------Method	4------------	------------>	
None	|| s_sheet | m_sheet	|| s_row/col | m_row/col	|| inspect_graph	|| s_sheet | m_sheet	
Display (s_sheet)	|| m_sheet	|| s_row/col | m_row/col	|| inspect_graph	|| m_sheet	2
Display (m_sheet)	|| s_sheet	|| s_row/col | m_row/col	|| inspect_graph	|| s_sheet	2
Display (s-row/col)	|| s_sheet | m_sheet	|| m_row/col	|| inspect_graph	|| s_sheet | m_sheet	1
Display (m_row/col)	|| s_sheet | m_sheet	|| s_row/col	|| inspect_graph	|| s_sheet | m_sheet	1
	Unit-task 5	**Unit-task 6**	**Unit-task 7**	**Unit-task 8**	
	<------------	------------Method	4------------	------------>	
None	|| s_row/col | m_row/col	|| inspect_graph	-	-	
Display (s_sheet)	|| s_row/col | m_row/col	|| inspect_graph	-	-	
Display (m_sheet)	|| s_row/col | m_row/col	|| inspect_graph	-	-	
Display (s-row/col)	|| m_row/col	|| inspect_graph	-	-	1
Display (m_row/col)	|| s_row/col	|| inspect_graph	-	-	1

<table>
<tr><th>Truncation</th><th>Unit-task 1</th><th>Unit-task 2</th><th>Unit-task 3</th><th>Unit-task 4</th><th>n_e-n_t</th></tr>
<tr><td></td><td colspan="4"><--<i>Method 5</i>--></td><td></td></tr>
<tr><td>None</td><td>|| s_sheet</td><td>|| s_row/col | m_row/col</td><td>|| inspect_graph</td><td>|| r[m_sheet | inspect_graph]</td><td></td></tr>
<tr><td>Display (s_sheet)</td><td>-</td><td>|| s_row | m_row/col</td><td>|| inspect_graph</td><td>|| r[m_sheet | inspect_graph]</td><td>1</td></tr>
<tr><td>Display (s_row/col)</td><td>|| s_sheet</td><td>|| m_row/col</td><td>|| inspect_graph</td><td>|| r[m_sheet | inspect_graph]</td><td>1</td></tr>
<tr><td>Display (m_row/col)</td><td>|| s_sheet</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>|| r[m_sheet | inspect_graph]</td><td>1</td></tr>
<tr><td>Task (inspect_graph)</td><td>|| s_sheet</td><td>|| s_row/col | m_row/col</td><td>-</td><td>|| r[m_sheet | inspect_graph]</td><td>1</td></tr>
<tr><td></td><td>Unit-task 5</td><td>Unit-task 6</td><td>Unit-task 7</td><td>Unit-task 8</td><td></td></tr>
<tr><td></td><td colspan="4"><--<i>Method 5</i>--></td><td></td></tr>
<tr><td>None</td><td>|| s_sheet</td><td>|| s_row/col | m_row/col</td><td>|| inspect_graph</td><td>|| r[m_sheet | inspect_graph]</td><td></td></tr>
<tr><td>Display (s_sheet)</td><td>-</td><td>|| s_row | m_row/col</td><td>|| inspect_graph</td><td>|| r[m_sheet | inspect_graph]</td><td>1</td></tr>
<tr><td>Display (s_row/col)</td><td>|| s_sheet</td><td>|| m_row/col</td><td>|| inspect_graph</td><td>|| r[m_sheet | inspect_graph]</td><td>1</td></tr>
<tr><td>Display (m_row/col)</td><td>|| s_sheet</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>|| r[m_sheet | inspect_graph]</td><td>1</td></tr>
<tr><td>Task (inspect_graph)</td><td>|| s_sheet</td><td>|| s_row/col | m_row/col</td><td>-</td><td>|| r[m_sheet | inspect_graph]</td><td>1</td></tr>
</table>

<table>
<tr><th>Truncation</th><th>Unit-task 1</th><th>Unit-task 2</th><th>Unit-task 3</th><th>Unit-task 4</th><th>n_e-n_t</th></tr>
<tr><td></td><td colspan="4"><--Method 6--></td><td></td></tr>
<tr><td>None</td><td>|| s_sheet | m_sheet</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>|| r[m_row/col | inspect_graph]</td><td></td></tr>
<tr><td>Display (s_sheet)</td><td>|| m_sheet</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>|| r[m_row/col | inspect_graph]</td><td>1</td></tr>
<tr><td>Display (m_sheet)</td><td>|| s_sheet</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>|| r[m_row/col | inspect_graph]</td><td>1</td></tr>
<tr><td>Display (s_row/col)</td><td>|| s_sheet | m_sheet</td><td>-</td><td>|| inspect_graph</td><td>|| r[m_row/col | inspect_graph]</td><td>1</td></tr>
<tr><td>Task (inspect_graph)</td><td>|| s_sheet | m_sheet</td><td>|| s_row/col</td><td>-</td><td>|| r[m_row/col | inspect_graph]</td><td>1</td></tr>
</table>

<table>
<tr><th></th><th>Unit-task 5</th><th>Unit-task 6</th><th>Unit-task 7</th><th>Unit-task 8</th><th></th></tr>
<tr><td></td><td colspan="4"><--Method 6--></td><td></td></tr>
<tr><td>None</td><td>|| s_sheet | m_sheet</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>|| r[m_row/col | inspect_graph]</td><td></td></tr>
<tr><td>Display (s_sheet)</td><td>|| m_sheet</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>|| r[m_row/col | inspect_graph]</td><td>1</td></tr>
<tr><td>Display (m_sheet)</td><td>|| s_sheet</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>|| r[m_row/col | inspect_graph]</td><td>1</td></tr>
<tr><td>Display (s_row/col)</td><td>|| s_sheet | m_sheet</td><td>-</td><td>|| inspect_graph</td><td>|| r[m_row/col | inspect_graph]</td><td>1</td></tr>
<tr><td>Task (inspect_graph)</td><td>|| s_sheet | m_sheet</td><td>|| s_row/col</td><td>-</td><td>|| r[m_row/col | inspect_graph]</td><td>1</td></tr>
</table>

<table>
<tr><th>Truncation</th><th>Unit-task 1</th><th>Unit-task 2</th><th>Unit-task 3</th><th>Unit-task 4</th><th>n_e-n_t</th></tr>
<tr><td></td><td colspan="4"><--Method 7--></td><td></td></tr>
<tr><td>None</td><td>|| s_sheet | m_sheet</td><td>|| s_row/col | m_row/col</td><td>|| inspect_graph</td><td>|| s_sheet</td><td></td></tr>
<tr><td>Display (s_sheet)</td><td>|| m_sheet</td><td>|| s_row/col | m_row/col</td><td>|| inspect_graph</td><td>-</td><td>2</td></tr>
<tr><td>Display (m_sheet)</td><td>|| s_sheet</td><td>|| s_row/col | m_row/col</td><td>|| inspect_graph</td><td>|| s_sheet</td><td>1</td></tr>
<tr><td>Display (s-row/col)</td><td>|| s_sheet | m_sheet</td><td>|| m_row/col</td><td>|| inspect_graph</td><td>|| s_sheet</td><td>1</td></tr>
<tr><td>Display (m_row/col)</td><td>|| s_sheet | m_sheet</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>|| s_sheet</td><td>1</td></tr>
<tr><th></th><th>Unit-task 5</th><th>Unit-task 6</th><th>Unit-task 7</th><th>Unit-task 8</th><th></th></tr>
<tr><td></td><td colspan="4"><--Method 7--></td><td></td></tr>
<tr><td>None</td><td>|| s_row/col | m_row/col</td><td>|| inspect_graph</td><td>|| r[m_sheet | inspect_graph]</td><td></td><td></td></tr>
<tr><td>Display (s_sheet)</td><td>|| s_row | m_row/col</td><td>|| inspect_graph</td><td>|| r[m_sheet | inspect_graph]</td><td></td><td></td></tr>
<tr><td>Display (s_row/col)</td><td>|| m_row/col</td><td>|| inspect_graph</td><td>|| r[m_sheet | inspect_graph]</td><td></td><td>1</td></tr>
<tr><td>Display (m_row/col)</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>|| r[m_sheet | inspect_graph]</td><td></td><td>1</td></tr>
<tr><td>Task (inspect_graph)</td><td>|| s_row/col | m_row/col</td><td>-</td><td>|| r[m_sheet | inspect_graph]</td><td></td><td>1</td></tr>
</table>

<table>
<tr><th>Truncation</th><th>Unit-task 1</th><th>Unit-task 2</th><th>Unit-task 3</th><th>Unit-task 4</th><th>n_e-n_t</th></tr>
<tr><td></td><td colspan="4"><--Method 8--></td><td></td></tr>
<tr><td>None</td><td>|| s_sheet | m_sheet</td><td>|| s_row/col | m_row/col</td><td>|| inspect_graph</td><td>|| s_sheet | m_sheet</td><td></td></tr>
<tr><td>Display (s_sheet)</td><td>|| m_sheet</td><td>|| s_row/col | m_row/col</td><td>|| inspect_graph</td><td>|| m_sheet</td><td>2</td></tr>
<tr><td>Display (m_sheet)</td><td>|| s_sheet</td><td>|| s_row/col | m_row/col</td><td>|| inspect_graph</td><td>|| s_sheet</td><td>2</td></tr>
<tr><td>Display (s-row/col)</td><td>|| s_sheet | m_sheet</td><td>|| m_row/col</td><td>|| inspect_graph</td><td>|| s_sheet | m_sheet</td><td>1</td></tr>
<tr><td>Display (m_row/col)</td><td>|| s_sheet | m_sheet</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>|| s_sheet | m_sheet</td><td>1</td></tr>
<tr><th></th><th>Unit-task 5</th><th>Unit-task 6</th><th>Unit-task 7</th><th>Unit-task 8</th><th></th></tr>
<tr><td></td><td colspan="4"><--Method 8--></td><td></td></tr>
<tr><td>None</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>|| r[m_row/col | inspect_graph]</td><td></td><td></td></tr>
<tr><td>Display (s_sheet)</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>|| r[m_row/col | inspect_graph]</td><td></td><td></td></tr>
<tr><td>Display (m_sheet)</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>|| r[m_row/col | inspect_graph]</td><td></td><td></td></tr>
<tr><td>Display (s_row/col)</td><td>-</td><td>|| inspect_graph</td><td>|| r[m_row/col | inspect_graph]</td><td></td><td>1</td></tr>
<tr><td>Task (inspect_graph)</td><td>|| s_row/col</td><td>-</td><td>|| r[m_row/col | inspect_graph]</td><td></td><td>1</td></tr>
</table>

<table>
<tr><th>Truncation</th><th>Unit-task 1</th><th>Unit-task 2</th><th>Unit-task 3</th><th>Unit-task 4</th><th>n_e-n_t</th></tr>
<tr><td></td><td colspan="4"><--<i>Method 9</i>---></td><td></td></tr>
<tr><td>None</td><td>|| s_sheet</td><td>|| s_row/col | m_row/col</td><td>|| inspect_graph</td><td>|| r[m_sheet | inspect_graph]</td><td></td></tr>
<tr><td>Display (s_sheet)</td><td>-</td><td>|| s_row | m_row/col</td><td>|| inspect_graph</td><td>|| r[m_sheet | inspect_graph]</td><td>1</td></tr>
<tr><td>Display (s_row/col)</td><td>|| s_sheet</td><td>|| m_row/col</td><td>|| inspect_graph</td><td>|| r[m_sheet | inspect_graph]</td><td>1</td></tr>
<tr><td>Display (m_row/col)</td><td>|| s_sheet</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>|| r[m_sheet | inspect_graph]</td><td>1</td></tr>
<tr><td>Task (inspect_graph)</td><td>|| s_sheet</td><td>|| s_row/col | m_row/col</td><td>-</td><td>|| r[m_sheet | inspect_graph]</td><td>1</td></tr>
<tr><td></td><th>Unit-task 5</th><th>Unit-task 6</th><th>Unit-task 7</th><th>Unit-task 8</th><td></td></tr>
<tr><td></td><td colspan="4"><--<i>Method 9</i>---></td><td></td></tr>
<tr><td>None</td><td>|| s_sheet | m_sheet</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>|| r[m_row/col | inspect_graph]</td><td></td></tr>
<tr><td>Display (s_sheet)</td><td>|| m_sheet</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>|| r[m_row/col | inspect_graph]</td><td>1</td></tr>
<tr><td>Display (m_sheet)</td><td>|| s_sheet</td><td>|| s_row/col</td><td>|| inspect_graph</td><td>|| r[m_row/col | inspect_graph]</td><td>1</td></tr>
<tr><td>Display (s_row/col)</td><td>|| s_sheet | m_sheet</td><td>-</td><td>|| inspect_graph</td><td>|| r[m_row/col | inspect_graph]</td><td>1</td></tr>
<tr><td>Task (inspect_graph)</td><td>|| s_sheet | m_sheet</td><td>|| s_row/col</td><td>-</td><td>|| r[m_row/col | inspect_graph]</td><td>1</td></tr>
</table>

Appendix 4

Preliminary data collection questionnaire

Bioview: initial assessment

Part 1: Assessment context

Name:	
Gender:	
School address	
Telephone number	

Please tick which of the following roles you are assuming in this selection exercise.

Prospective user of the software in teaching a subject, e.g.biology.	
Assessor of the software with an open view on its eventual use, e.g. a school IT coordinator.	

Please describe the context in which you are assuming the software is going to be used.

Bioview assessment exercise: Design and evaluation of educational software (7.5.91)

Part 2: Familiarisation

Please describe your reactions to the software as you familiarised yourself with the way the software operated.

Part 3: Assessment of classroom use

What curriculum areas do you think the software could be relevant to? Please explain your comments.

Bioview assessment exercise: Design and evaluation of educational software (7.5.91)

Do you think the software would be accessible to the students you teach or you imagine being taught by other teachers? Please give reasons for your answer.

What do you think the benefits of using this software would be? Please explain your comments.

Bioview assessment exercise: Design and evaluation of educational software (7.5.91)

Part 4: User interface design

What do you think the potential benefits and weaknesses of the user interface design would be in the use of the software by pupils?

Part 5: Any other comments

Please make any other comments you feel are appropriate.

Bioview assessment exercise: Design and evaluation of educational software (7.5.91)

Appendix 5

An example of an episode record sheet

Episode R4: Ruth/Tom session, Episode 4

Top-level goal

To use the program to answer Question 3 (Try to find either the optimum value of light intensity or carbon dioxide). Tom used the program with advice from Ruth to find the optimum value of the light intensity.

Method: 1(T)

Action string

| s_sheet(T) | m_row(C) | m_row(C) | scope_win | m_row(C) | inspect_graph(r/L)
|{s_sheet(T)} | m_sheet(T) | inspect_graph(r/L) | s_graph(l) | s_graph(b)
| inspect_graph(r/L)

Unit task 1

To explore the effect on the rate of photosynthesis of varying the light intensity

Technique

Execute m_row(C) operations to maximise the the carbon dioxide level at the optimum temperature, and inspect the corresponding rate/L graph.

Observed action substring

|| s_sheet(T) | m_row(C) | m_row(C) | scope_win | m_row(C) | inspect_graph(r/L)

Corrected action substring

|| s_sheet(T) | m_row(C) | m_row(C) | scope_win | m_row(C) | inspect_graph(r/L)

Reduced action substring

|| s_sheet(T) | m_row(C) | inspect_graph(r/L)

Expert action substring

|| s_sheet(T) | m_row(C) | inspect_graph(r/L)

System signature

(i) Change in the T-sheet sub-register: intermediate C-value changed to a maximum C-value

Display signature

(i) Change in the datacube display window: T-sheet displayed instead of L-sheet

(ii) Change in the graph_window(2) display: an instance of a rate/L graph displayed instead of rate/T graph

Transcript

T: Try to find optimum level of either light or carbon dioxide.

R:You can do either one or can do both.

T: So to find this we need to have light or carbon dioxide along this side, [*referring to the vertical axis of the rate/T graph displayed in graph window(2)*] which we've got anyway.

R: No, we'd need light or carbon dioxide along here. [*referring to the horizontal axis of the rate/T graph displayed in graph window(2)*]

T: Along there? Oh I see, right.

R: Along the bottom, so basically the same as finding same level of temperature. So what we need to do is, need to put temperature. If you click on temperature [*s-sheet(T) operation executed*] so got light intensity over here. [*referring to the horizontal axis of the rate/T graph displayed in graph window(2)*]

T: I see, temperature's fixed now at 30. [*referring to the T-value display in graph window(2)*]

R:Yeah.

T: If I try moving along here, shall I? [*m-row(C) operation executed*] Alright, go other way. [*m-row(C) operation executed*]

R:Yeah, need to go back.

T: Over here? [*scroll_win operation executed*]

R: [*m-row(C) operation executed to move the C-row to the maximum position*] Yeah. 0.1.

T: Is that the highest point we can go? [*referring to the maximum C-value for the displayed row*]

R: Yeah, that's maximum there. [*pointing to the rate value for the maximum L-value shown on the horizontal axis of the rate/L graph*]

T: Would seem to be here.

Unit task 2

To maximise the temperature

Technique

Locate the T-sheet in the maximum T-value position by executing a m_sheet(T) operation

Observed action substring

||{s_sheet(T)} | m_sheet(T) | inspect_graph(r/L) | s_graph(l) | s_graph(b)

Corrected action substring

|| m_sheet(T) | inspect_graph(r/L) | s_graph(l) | s_graph(b)

Reduced action substring

|| m_sheet(T) | inspect_graph(r/L)

Expert action substring

Not applicable

System signature

(i) Change in T-sheet sub-register: change in T-value from an intermediate value to the maximum value

Display signature

(i) Change in the datacube display window: T-sheet corresponding to the maximum T-value displayed

(ii) Change in graph_window(2): bar graph selected temporarily, but a line graph was shown at the end of the unit-task.

Transcript

R:Yes, haven't got maximum temperature so ...

T: If I go back here. [*referring to the T-value scroll box*]

R:Yeah

T: Click that [*redundant s_sheet(T) operation executed*] and increase temperature like that. [*m_sheet(T) operation executed*] Right, I think 40 is the maximum. So this one here ... [*referring to the rate value corresponding to the maximum L-value*]

R:Yeah so that is the ...

T: Is there any way we can actually. Hold on. [*sequence of s_graph(l) and s_graph(b) operations executed*] I'm wondering if there is any function on this that allows you to look at maximum and minimum values without having to ... Because I can't see that easily anyway [*indicating a difficulty in reading the maximum rate value form the displayed graph*]. All these here to me look the same.

R: Difficult to tell, but 50 is max.

Unit task 3

To determine the optimum value of the light intensity

Technique

Inspect the existing rate/L graph

Observed action substring

|| inspect_graph(r/L)

Corrected action substring

|| inspect_graph(r/L)

Reduced action substring

|| inspect_graph(r/L)

Expert action substring

|| inspect_graph(r/L)

System signature

No change

Display signature

No change

Transcript

R: Yeah so that is the ... [*indicating that the optimum value of the light intensity is now shown on the rate/L graph*]

Comments

In Unit-task 1 Ruth suggested that finding the optimum value of light or carbon dioxide was "basically the same as finding same level of temperature". This implied that she thought that the same method she used in Episode R2 to determine the optimum temperature (Method 1) could be used to determine the optimum value of the light intensity or carbon dioxide level. For example, applying Method 1 to determine the optimum value of the light intensity would require both the C-value and the T-value to be maximised (executing an appropriate combination of m-sheet and m_row/col operations) followed by an inspection of the resulting rate/L graph. However, in this case the temperature should not be at the maximum value; it should be at the optimum value.

The goal structure indicated that Ruth was recommending Method 1 to find the optimum temperature. The sub-goal of Unit-task 1 was to maximise the C-value at the optimum temperature by executing an m_row(C) operation on a T-sheet located at the optimum temperature location. In Unit-task 2 the optimum temperature was changed to a maximum value by executing an m_sheet(T) operation. In Unit-task 3 the rate/L was inspected again in order to determine the optimum value of the light intensity with the C-value and the L-value both fixed at a maximum.

Note that that there was some repetition in Unit-task 2 and Unit-task 3 - both involved an interpretation of the displayed rate/L graph to identify the optimum value of the light intensity. If Method 1(L) had been applied without error the graph would only have been interpreted at the end of Unit-task 3. (Error here applies strictly to the implementation of a chosen method; not to any "cognitive error" associated with a misinterpretation of the task).

There was a confusion between the idea of a maximum and an optimum value. Was this caused by the description of the task in Question 1 ("make sure that neither light or carbon dioxide are in limited supplies")? Ruth may have thought that "unlimited" was synonymous with "maximum".

Viewed in isolation (that is, as a complete episode) Unit-task 1 could be interpreted as an example of the use of Method 3 (on the T-sheet corresponding to the optimum temperature execute m_row(C) operations and observe the effect on a rate/L graph to determine the optimum value of the carbon dioxide level). This illustrates the importance of defining episode boundaries from a task perspective, as opposed to a system perspective.

The first part is relevant to Tom's misunderstanding of the relationship between the datacube representation and the graph displays (also evident in Episode R2). Ruth demonstrated that she did not share this misunderstanding.

Appendix 6

Laboratory session descriptions

Delia/Alice session

This session consisted of 14 episodes. The first seven episodes featured demonstrations by Delia of how to use *Bioview.* All but the first of these episodes were focused on the use of the software to answer one of the worksheet questions; the first episode consisted of a demonstration of the basic features of the software. In the remaining seven sessions Alice was invited by Delia to use the program under her supervision. Each of these episodes was concerned with answering a specific worksheet question.

Episode D1

This episode was a "warm-up" exercise in which Delia introduced Alice to the basic operation of the datacube and adjustment of the visible scope of the datasheet window. She did not demonstrate how to link graphs to the datacube and datasheet windows.

> D: On the screen is some data and it shows light intensity, temperature and carbon dioxide, and all compacted onto a cube model. [*pointing to the screen*] And you can increase temperature, and you can increase or decrease carbon dioxide and light intensity by actually bringing the arrow up to this bit, [*T-scroll box*] and just clicking [*m_sheet(T) operation executed*].
>
> A: Now made it smaller.
>
> D: Let's increase the temperature. To increase temperature you move red bit across up there like that. [*referring to the datasheet T-value indicator*] If you want to decrease temperature you actually go to arrow going towards left. [*m_sheet(T) operation executed*] Just click on it to wherever you want to go to.
>
> A: Right
>
> D: And same with carbon dioxide. [*s_sheet(C) operation executed*]
>
> A: So down.
>
> D: Carbon dioxide down is increase, [*m_sheet(C) operation executed*] up is decrease, [*m_sheet(C) operation executed*] and for light intensity ... Whichever one you click on the keys in the corner over there. [*referring to the datasheet value displayed in the top left hand corner of the datasheet*] For light intensity [*s_sheet (L) operation executed*] left is increase and right is decrease, [*attempt to execute m_sheet(L) operations in the wrong directions*] no it isn't, left is decrease and right is increase, yeah? [*m_sheet L) operations executed in the*

right directions] So this is maximum and this is minimum. Yeah, do you get that?

A: Yeah

D: The sheet actually shows all the different possible combinations there, got the temperatures across there, carbon dioxide is up there, and you can have a set, light intensity as well, [*gesturing with the hand and mouse to various parts of the datasheet*] and you can use the data to actually answer questions on the sheet. If you want to see what's on other ... You want to decrease temperature up here, you'll see. Just press there, [*scope_win operation executed for left-hand window scroll box*] and if you want to increase it press there. [*scope_win operation executed for right-hand window scroll box*] And the same with this bit really. [*scope_win operation executed for vertical window scroll boxes*] Yeah? OK. Graph isn't connected to data as yet [...].

Episode D2

In this episode Delia successfully used Method 1(C) to demonstrate how to use the program to find the optimum temperature for the rate of photosynthesis. However, the attempt to maximise both the light intensity (Unit-task 1) and the carbon dioxide level (Unit-task 2) by executing m_sheet operations indicates that Delia held the "active system sub-register misconception" as described in Section 7.1.3.1. A graph window was only connected in Unit-task 3, so the effects of the datacube manipulations could not be observed until the final stages of the application of the method. A bar chart was chosen to make it easier to identify the optimum temperature. In Unit-task 3 the s_row(T) operation and the following scope_win operations were executed in error. This can be interpreted as an attempt to change the column position even though a row was highlighted. In this sense the following s_col(L) operation was an undo operation corresponding to the previous s_row(T) operation.

Unit-task 1

D: OK. Graph isn't connected to data as yet, and first question is, have to use data, says optimum air temperature at maximum rate of photosynthesis and you have to make sure that light or carbon dioxide are not in limited supply, so basically got to be unlimited. So what you have to do is you go to temperature, [*s_sheet(T) operation executed*] and .. No you go to light intensity [*s_sheet (L) operation executed*] and put it to maximum. [*m_sheet (L) operation executed*]

Unit-task 2

A: And the same with carbon dioxide.

D: Go to carbon dioxide [*s_sheet(C) operation executed*] and put that up to maximum as well. [*m_sheet(C) operation executed*]

Unit-task 3

D: Graph type, what sort of graph do you think it will be?

A: Line one.

D: OK, right. [*s-graph(l) operation executed*] Just have a look. Display. [*s-row(T) operation executed in error*] If we go down [*scope_win operation executed in an attempt to undo the execution of the previous s_row(T) operation*] to the light intensity there, [*s_col(L) operation executed successfully to undo the execution of the s_row(T) operation*] there you've got maximum carbon dioxide, got maximum light intensity, yeah, and got temperature and rate of photosynthesis. [*referring to the axes shown in the graph_window(1)*] [*m_col(L) operation executed to maximise the value of the light intensity*] Now you just go to connect and connect graph. [*con-graph(1) operation executed*] Just look at the graph, [*interpret the displayed rate/T graph*] and where on the graph do you think is photosynthesis the greatest?

A: 30 degrees.

D: Yeah. If you try you can change the type and go to bar graph, [*s-graph(b) operation executed*] and again you can see its 30 degrees. [*inspect_graph(r/T) operation executed*]

Episode D3

In Question 2 the learners were invited to look at the way varying the light and carbon dioxide levels affected the rate of photosynthesis. This was perceived by Delia as two separate tasks, that is, one task concerned with the effect of varying the light intensity (Episode D3A), and one task concerned with the effect of varying the carbon dioxide level (Episode D3B).

Episode D3A

This episode featured an execution of Method 3(T). The errors in Unit-task 3 were probably caused by the display of a null rate/C graph as discussed in Section 7.1.2. The technique used in Unit-task 1 to create a second graph window indicates that Delia was not very familiar with *Windows*.

Unit-task 1

OK, second question - at this optimum temperature look at the way in which varying the light or carbon dioxide levels affects the rate of photosynthesis. For this you could have two graphs so, start by going to graphs [*start_graph operation executed*] and then pull that one [*two resize_win operations executed*] across, because the other one is under it.

Unit-task 2

So what you do, you have to get it to optimum temperature, [*s_sheet(T) operation executed*] which is 30 degrees, so that's 30 degrees, [*m_sheet(T) operation executed*] and then ...

A: How do you know that?

D: Because what you do is, when you increase temperature [*Delia pointing with her hand to indicate the current T-sheet position*] in the corner that's 30 degrees, [*Delia pointing with her hand to indicate the current T-sheet value displayed in the datasheet window*] isn't it?

A: Yeah

D: You understand? 30 degrees up there.

Unit-task 3

Right, first of all you do light intensity and connect this [*three redundant con_graph(1) operations executed*] display; it's not connecting, [*select_col(L) operation executed*] Connect. [*redundant con_graph(1) operation executed*] OK [*m_col(L) operation executed*]. Light intensity is five, yeah? Now if we move along, light intensity 10. Lets see what happens, it's not connecting, to rate of photosynthesis. Do line graph [*s_graph(l) operation executed*]. Watch, 10 [*m_col(L) operation executed*] is that much, increase light intensity to 15, [*scope_win operation executed*] its that much, 20, 25. [*m_col(L) operation executed*]

A: So it increases. [*scope_win and m_col(L) operations executed*]

D: What's happening?

A: The rate of photosynthesis is increasing.

D: Hang on a minute. [*scope_win and m_col(L) operations executed*]

A: Increase light. [*scope_win and m_col(L) operations executed*]

D: Right, 15, 20, 25, 30 [*scope_win and m_col(L) operations executed*], 35 it increases, 40 it increases, 45 ... Does it go down then? [*scope_win operation executed*]

A: Yeah went down [*m_col operation executed*] 45, 50.

D: So overall generally as you increase light intensity, rate of photosynthesis increases as well.

Episode D3B

This episode also featured an application of Method 3(T). In Unit-task 1 Delia thought that the value of the carbon dioxide concentration could be changed by clicking on the C-value scroll box. She was confusing changing the C-value associated with a datasheet and the C-value associated with a column. She acknowledged her confusion - "Now if we display carbon dioxide. Bear with me." - and reset the display by clicking on the T-value scroll box to show the T-sheet. The s_col(L) operation executed in error was probably to confirm that she had reset the state of the display properly. This operation was undone by executing an s_row(C) operation. In Unit-task 2 graph_window(2) was connected and m_row(C) and scope_win operations were executed to set the C-value at it minimum row position, followed by the execution of m_row(C) and scope_win operations to observe the effects of varying the value of the carbon dioxide concentration. It is not clear why Delia connected a second graph as she did not make a comparison between the two graph window displays.

Unit-task 1

> D: [...] Now if we display carbon dioxide amount. [*s_sheet(C) operation executed*] Bear with me. Temperature at 30 [*s_sheet(T) operation executed*] and carbon dioxide down here, [*indicating an L-column*] so display columns, [*s_col(L) operation executed in error*] rows. [*s_row(C) operation executed to undo the previous s_col(L) operation*] Sorry.

Unit-task 2

> D: Connect. [*con_graph(2) operation executed*] This is ... get right to the top. [*sequence of m_row(C) and scope_win operations executed to set the C-value at minimum*] Right, at 0.01. [*after executing m_row(C) and scope_win operations to observe the effect of changing the C-value*] Do you want to just talk through the graphs?
>
> A: As increase amount of carbon dioxide the graph seems to, rate of photosynthesis seems to be increasing.
>
> D: Yeah, so that's that question.

Episode D4

Question 3 invited the learners to determine either the optimum value of light intensity or carbon dioxide concentration. In this episode the optimum value for both was determined. Episode D4A featured the determination of the optimum value of the light intensity, and Episode D4B featured the determination of the optimum concentration of carbon dioxide.

Episode D4A

Method 1(T) was successfully applied to identify the optimum value of the light intensity by inspecting the existing rate/L graph shown in the currently connected graph_window(2). This graph corresponded to the optimum temperature (the appropriate sheet was chosen in the previous episode as part of the answer to Question 2) and the maximum value of carbon dioxide (due to the current row position).

Unit-task 1

D: Number three, try to find optimum level of either light or carbon dioxide. Which one do you want to do?

A: Light.

D: Right. Basically if you do it type, [*referring to the type menu item in graph_window(2)*] do bar graph for that one. [*s_graph(b) operation executed*] Like rate, where rate of photosynthesis is the greatest, so if look at the graph the light intensity is increasing along the bottom, and where is rate of photosynthesis the greatest?

A: About 50 degrees. [*inspect-graph(r/L) operation executed*]

D: 50?

A: Yeah

D: So that would be the optimum level because its rate of photosynthesis is greatest at that level of light.

Episode D4B

The optimum value of the carbon dioxide concentration was determined by applying Method 3(T) to observe the effect of changing the value of the carbon dioxide concentration on the rate/L graph displayed in graph_window(1). Again a bar chart was preferred to a line graph.

Unit-task 1

D: Do you want to do carbon dioxide as well?

A: Yeah, might as well.

D: Here. Make it into bar graph. [*referring to the graph shown in graph window(1) and a s_graph(b) operation executed*]. Right. This is at point one, [*0.1*] on this graph carbon dioxide is up there, [*pointing to the value of the carbon dioxide concentration displayed on the top display bar of graph window (1)*] yeah, and if I change it what happens? [*m_row(C) operation executed*] Connect it. [*con_graph(1) operation executed*] Point 9. [*0.09*] Decrease carbon dioxide, [*m_row operation executed*] so optimum amount is actually point one [*0.1*] because you're getting greatest rate of photosynthesis at that amount. [*inspect_graph(r/L) operation executed*]

Episode D5

This episode was concerned with the use of the program to answer Question 4, that is to see if there is any difference in the effect of increasing the levels of of light intensity and carbon dioxide concentration on the rate of photosynthesis. In order to do this Delia successfully applied Method 6(TT) in an error free fashion. The effect of varying the carbon dioxide concentration was explored in Unit-task 1, and the effect of changing the light intensity was considered in Unit-task 2. The connection of graph_window(2) in Unit-task 1 was unnecessary as both graph_window(1) and graph window(2) displayed rate/L graphs at the start of the episode. Delia did not appear to use the concept of limiting factors, that is to identify which of the factors was closest to its non-limiting value.

Unit-task 1

> D: Is there any difference in the effect of increasing levels of light or carbon dioxide on the rate of photosynthesis? You could look at the graphs and look at the difference in the effect of increasing light and increasing carbon dioxide, [*referring to the two rate/L graphs displayed on the screen*] so if you look at this graph again, and go back to this one. [*scope_win and m_row(C) operations executed to initialise the C-value at the minimum value, followed by the execution of a con_graph(2) operation*] So increase, increase. [*m_row(C) and scope_win operations executed; s_graph(l) operation executed to select a line graph*] Right, it increases quite quickly. [*inspect_graph(r/L) operation executed*]

Unit-task 2

> D: Now if we look at light intensity, say this one, display columns instead, [*s_col(L) operation executed*], and go that way [*m_col(L) and scope_win operations executed*]. Right, this is light intensity one. So we know it increases anyway as increase that. [*the value of the light intensity*] But does it increase more dramatically than carbon dioxide? [*inspect_graph(r/C) operation executed*]
>
> A: No, seems a much smaller change.
>
> D: So if you increase carbon dioxide the rate of photosynthesis is more, changes more quickly than with light intensity.

Episode D6

In this episode Question 5 was answered in a very superficial way without using *Bioview*. Neither Delia or Alice appeared to be aware of the concept of limiting factors.

Unit-task 1

D: OK. Number 5 - at the fastest rate of photosynthesis shown in the data, which factor would you try to increase more to attempt to increase the rate further. How do you think you could do that?

A: You could have a look at both carbon dioxide and light intensity and see where the rate of increase of photosynthesis is at the most, and you see which one effects photosynthesis the most.

D: Yeah. [...]

Episode D7

This episode featured the first attempt by Alice to use *Bioview* with advice from Delia. (This episode took place before researcher intervention to inform the learners of a display bug). An attempt to answer Question 1 formed the task focus. As in Episode D2 Delia thought that the carbon dioxide concentration and the light intensity could both be maximised by executing m_sheet operations; further evidence of the active system sub-register misconception. However, in this episode Method 3(L) was successfully applied after the two initial m_sheet operations, in contrast to the application of Method 1(C) in Episode D2.

Unit-task 1

D: Do you want to have a go at the questions? Swap over.

A: So first thing I do, first go to the cube.

D: Yeah. Find optimum air temperature for the maximum rate of photosynthesis.

A: Do I press this button? [*pointing with the mouse to the C-value scroll box*]

D: Well first of all you've got to make sure that neither light nor carbon dioxide are in limited supplies, so go to light and ... Yeah, just increase for maximum [*confirming that Alice should execute an s_sheet(C) operation (so as to start to implement a method intended to locate the C-sheet at the maximum C-value position) before attempting to maximise the L-value*].

A: Just press it. [*C-value scroll box*]

D: Yeah. [*s_sheet(L) operation executed*] So this shows that that's maximum at point 1, [*0.1*] yeah?

Unit-task 2

A: And I do the same with this one? [*referring to the L-value scroll box*]

D: Light intensity? Yeah. [*s_sheet(L) operation executed*] That's maximum as well. Now have to connect data to the graph and press connect. [*connect_graph(1) operation executed*] OK. If you go to that one, [*referring to a T-column*] take it to the end. [*scope_win)operation executed*] Is that as far as it can go?

A: Yeah

D: OK now click on this. [*m_col(T) operation executed*] Now just go across and see where it increases most. Go right across the temperature range.

Episode D8

A second attempt was made by Alice in this episode to answer Question 1. (This episode involved researcher intervention to inform the participants of a display bug). The episode featured an application of Method 2(T). In Unit-task 1 Delia told Alice to maximise the C-value by executing an m_row(C) operation, which implied that, if she was intending to advise the selection of Method 2(T), she would instruct Alice to inspect instances of a rate/L graph as the T-sheet location was changed. However, in Unit-task 2 she instructed Alice to execute m_col(L) operations to observe the effect of varying light intensity as shown by changes in a rate/C graph. In Unit-task 3 she realised her error and advised Alice to execute m_sheet(T) operations to observe the effect of changing temperature on a rate/C graph. The m_col(L) operations executed in Unit-task 2 left the L-value at an intermediate value for the T-sheet; thus the instances of rate/C graphs observed as the T-sheet was moved corresponded to a non-maximum L-value.

Unit-task 1

Researcher: Can I interrupt for a moment? There's a bug in the program there. Really if you look at that, what you need at the moment is carbon dioxide at the bottom. Need to change that to light intensity, properly.

D: Swap ... ? [*s_sheet(C) and s_sheet(T) operations executed*]

Researcher: Well, when you swap to columns should swap from carbon dioxide to light intensity. So as you had it before it should have read carbon dioxide down the bottom, should have read for column carbon dioxide and for a row light intensity. Go back to where you were before, change that to a row instead of a column.

D: If you go to display, go to row, click on row. [*s_row(C) operation executed*]

Researcher: Now that should read light intensity on the bottom. Got confused.

D: OK, so go to point 1 [*0.1*] cos got to have it in unlimited supplies [*of carbon dioxide*] and ...

A: Maximum? [*m_row(C) operation executed*]

Unit-task 2

D: Just go to display and to column, [*s_col(L) operation executed*] go back here [*scope_win operation executed to display the column corresponding to the minimum L-value*], right, basically go across [*m_col operation executed*] and see how graph increases and decreases. [*interpret-graph rate/C graph*]

A: Do I press this one for it to go across? [*the right hand horizontal window scroll box*]

D: Yeah. [*scope_win and m_col(L) operations executed*]

Unit-task 3

D: Oh! no you've got the temperature at 30, its temperature you're supposed to be .... If you click on there [*T-value scroll box*]

A: So

D: Just change that, [*the T-value corresponding to the T-sheet*] only goes up to 40, the temperature. So if you decrease again and just look and see where it increases and where it decreases because increasing temperature. [*explore the effect of changing temperature values by executing m_sheet(T) operations*]

A: Increasing

D: You can see up here [*T-value display in top menu bar*] what temperature you've got as increase and decrease it, yeah, you understand that bit?

A: So as I change temperature shows.

D: Just click. If change temperature, if go down to 10 temperature. Right now just increase from there, 20, 25, 30.

A: This is coming down. [*m_sheet(T) and inspect_graph(r/C) operations executed*]

D: Yeah, so its 30, cos jumps down from there, you understand?

A: Yeah.

Episode D9

This episode was concerned with Question 2. The effects of changing the light intensity were explored in Episode D9A, and the effects of changing the level of carbon dioxide were considered in Episode D9B.

Episode D9A

In this episode Delia instructed Alice to carry out Method 3(T) to explore the effect of changing the light intensity. In Unit-task 1 Delia verbally confirmed that the temperature needed to set at the optimum value. The datasheet was already located in this position but Alice incorrectly selected the C-sheet and executed an m_sheet(C) operation. Delia helped Alice to reset the datasheet location to the optimum temperature position. In Unit-task 2 Delia explained to Alice how to vary the value of the light intensity using m_col(L) operations, with the aim of observing changes in a sequence of rate/C graphs. There were four instances of redundant execution of con_graph(1) operations in this unit-task.

Unit-task 1

D: OK, number two. At this optimum temperature look at the way in which varying the light and carbon dioxide levels affects the rate of photosynthesis. So for this one you've got to have optimum temperature.

A: Which is 30 degrees.

D: Yeah

A: So do I just leave it like this? [*referring to the graph window(1) display*]

D: Yes, leave it at 30 degrees, but got to vary light and carbon dioxide.

A: So got to change it here? [*s_sheet(C) operation executed by clicking on C-value scroll box and executing an m_sheet(C) operation*]

D: So, if you go back to that, [*s_sheet(T) operation executed*] this is set at 30, yeah

Unit-task 2

D: Now if you go to that arrow here, [*left hand window scroll box*] take it to the end, [*scope_win operation executed to display the minimum L-value column*] now this is light intensity, connect graph. [*redundant con_graph(1) operation*] Go to display, [*display menu item not selected - the mouse click is off target and an m_col operation is executed*] now connect. [*redundant con_graph(1) operation*] OK go to that. [*m_col((L) operation executed to move to the second column*] Connect the graph [*redundant con_graph(1) operation*] and just carry on. [*m_col operations executed*] It's connected, it's connected. [*in response to Alice trying to execute a redundant con_graph(1) operation*] What you have to do is just increase, this is light increasing.

A: So just move it along [*rate/C graph inspected as scope_win and m_col(L) operations executed*]

D: Yeah, and just see what happens as you increase light intensity.

A: Seems to be increasing. [*the rate of photosynthesis*]

Episode D9B

In this episode Delia explained how to vary the value of the carbon dioxide concentration using m_row(C) operations, with the aim of observing changes in a sequence of rate/L graphs. Delia clearly understood the goal structure for Method 3(T). When Alice attempted an illegitimate m_col(L) operation Delia immediately corrected her, and informed her that "no this is carbon dioxide".

Unit-task 1

D: Yeah? OK do same for carbon dioxide levels.

A: So I have to change the display first?

D: Yeah.

A: To rows. [*s_row(C) operation executed*]

D: Yeah, good, now you need to go to the top. [*inappropriate scope_win operation executed - left hand horizontal window scroll box clicked on when the top right hand vertical window scroll box should have been clicked on*]

Researcher: So that should read light intensity at the bottom.

D: OK

A: Now go across? [*illegitimate attempt to execute m_col operation*]

D: No, you need to go to the top, this is carbon dioxide [*illegitimate attempt to execute m_col(L) operation corrected by Delia*] and you have to click up there [*minimum C-value row location*] to get right to the top. Now take it down and see how the graph changes. [*m_row(C) operation executed*] OK just bring it down [*scope_win and m_row(C) operations executed*] - so do you notice? [*interpret the changes in the rate/L graph*]

A: Seems to be increasing as you increase amount of carbon dioxide

Episode D10

This episode featured an attempt by Alice to identify the optimum light intensity in answer to Question 3. Delia provided "expert" instruction on how to use Method 3(T). The first operation was curious. Alice appeared to correctly execute an s_col(L) operation by selecting the appropriate menu item from the display bar in the datasheet window. However, not only was a column selection effected, but an m_col(L) operation which moved to the column corresponding to the minimum L-value was also effected. Perhaps

Alice double clicked, with the second click occurring just after the fall down menu for display had cleared, making her unconsciously execute an m_col(L) operation.

Unit-task 1

> D: And it says you've got to now find an optimum level of either light or carbon dioxide, so we decided to do light, so ...
>
> A: Change display first to columns. [*s_col(L) operation executed*]
>
> D: Yeah, OK, increase the [*light intensity*]. Just see where rate of photosynthesis is the greatest. [*m_col, scope_win operations and inspect_graph(r/C) operations executed*]
>
> A: 50 degrees.
>
> D: Yeah, and 50 light intensity, [*Delia corrects Alice's incorrect use of units*] yeah?
>
> A: Yeah.
>
> D: So that would be optimum, do you know why?
>
> A: Because at point where makes most photosynthesis.
>
> D: Yeah, where rate is greatest.

Episode D11

Delia advised the application of Method 6(TT) in this episode to see if there was any difference in the effect of increasing the levels of light or carbon dioxide on the rate of photosynthesis. Unit-task 1 was concerned with the effect of varying light intensity and Unit-task 2 was concerned with the effect of varying the level of carbon dioxide. In Unit-task 3 Delia gave repeat instructions on how to investigate the effect of varying light intensity. This repetition was due to the effects of a display bug which resulted in incorrect labelling of the horizontal graph axes.

Unit-task 1

> A: Do you see any difference in the effect of increasing levels of light or carbon dioxide. When doing Question 2 and increasing the light and then increase carbon dioxide, did you notice any difference in the rate of photosynthesis?
>
> D: I think it was ...
>
> D: Can do it in two graphs if you want and compare them both together. [*referring to the comparison of a rate/L graph and a rate/C graph displayed side by side*]

A: So take it [*the position of the L-column*] back first [*scope_win operation executed to to display the L-column corresponding to the minimum L-value*]

D: Right, OK, if you connect to this one. [*referring to graph_window(1)*] So if go to row first, [*m_col operation executed and minimum L-col - row not selected*], and connect. [*redundant con_graph(1) operation executed*]. Right OK start increasing light intensity. [*m-col and scope_win operations executed in sequence*]

A: Increases by fair amount each time.

Unit-task 2

D: So if you now change the display.

A: Its columns, no, rows. [*an attempt to execute an s_col(L) operation in error before executing an s_row(C) operation*]

D: Yeah, connect it while graph still connected. [*irrelevant scope-win operation execute (left hand horizontal window scroll box clicked on when columns selected) followed by a con_graph(2) operation*] OK, that's at point 1, [*0.1*] now if you go to minimum [*scope_win operation executed to display minimum C-row*], *yeah that's it, just bring it down from there* [*m_row(C) and scope_win operations executed to inspect changes in instances of rate/L graphs*].

Unit-task 3

A: Wouldn't be better if I used the same graph for both of them?

D: Yeah, you had light intensity one connected. [*referring to graph_window(1) which shows a rate?C graph when a rate/L graph should have been shown*]

A: They're both carbon dioxide.

D: Yeah, these are both carbon dioxide. If you hold on a minute. [*scope_win and m_row(C) operations executed*] OK, go back to this one. [*referring to graph_window(1) and con_graph(1) operation executed*] Press display, connect it with that one, [*referring to graph_window(1)*] connect with this first one. [*redundant con_graph(1) operation executed*] Display again, do columns. [*s_col(L) operation executed*] Now change to line graph.

A: Type? [*s_graph(l) operation executed*]

D: Yeah, OK start. [*m_col(L) operation executed to locate L-column at the minimum L-value position, followed by a series of m_col(L) and scope_win operations executed to inspect the displayed rate/L graph*]

A: Light intensity seems to be increasing. Slower than carbon dioxide.

D: Yeah, because this starts increasing from the origin. [*referring to the rate/L graph displayed in graph window(2)*]. This doesn't start until it's there [*referring to the rate/C graph which is displayed in graph window(1), pointing out that the first two C-values are zero*]

Episode D12

In this episode previous experience was recalled to try to answer Question 5 and identify at the fastest rate of photosynthesis which factor should be increased to increase the rate further. Both Delia and Alice appear not to be aware of the concept of limiting factors.

Unit-task 1

> D: [...] Right, at the fastest rate of photosynthesis shown in data, which factor would you try to increase more to attempt to increase the rate further. So at the fastest rate, which would be 36, [*pointing to the appropriate datasheet cell*] yeah, which factor would you try to increase more to increase rate?
>
> A: What, between carbon dioxide and light intensity?
>
> D: Yeah, which one do you think? If you use ...
>
> Researcher: That one should read light intensity. [*referring to the horizontal graph axis*].
>
> D: Yeah that's light intensity.
>
> A: Would increase light intensity.
>
> D: So could you show that on graph, can you show it on graph?
>
> A: Um...
>
> D: Would you be able to read off scale and see which one seems to be more? [*pointing with the mouse to graph window(1) - rate/C graph*]
>
> A: Yeah.
>
> D: So if you, basically its factor which causes the biggest change in the rate of photosynthesis which was, carbon dioxide? [*recall from previous episodes*]
>
> A: Yeah.
>
> D: Yeah? So probably going to be that factor, OK
>
> A: Yeah

Sharon/Uri session

This session consisted of 14 episodes. The first seven episodes featured demonstrations by Sharon of how to use *Bioview*. All but the first of these episodes were focused on the use of the software to answer one of the worksheet questions; the first episode consisted of a demonstration of the basic features of the software. In the remaining seven sessions Uri was invited by Sharon to use the program under her supervision. Each of these episodes was concerned with answering a specific worksheet question.

Episode S1

During this illustration of basic features there was clear evidence that Sharon understood the effect on the sheet-value variable of executing m_sheet operations.

> S: This is the constant at the moment [*pointing to the C-value shown in the sheet value box displayed in the datasheet window*] and you can change what the constant is. [*m_sheet(C) operations executed*] Do it like this, [*s_sheet(L) operation executed*] light intensity is near constant, you can change that, [*m_sheet(L) operations executed*] different rates, different temperatures, and carbon dioxide levels.

Episode S2

Sharon's execution of Method 1(L) in this episode indicated that she understood the sub-goal structure of this method. In Unit-task 1 she realised that the carbon dioxide level should be maximised and executed an m_row(C) operation on the currently displayed L-sheet to change the C-value from an intermediate value to a maximum value. In Unit-task 2 she was confused about her role as a teacher and indulged in a peripheral demonstration of how to manipulate the datacube to change the active datasheet. This led to the execution of arbitrary s_sheet(T) and s_sheet(L) operations. However, when she properly engaged in her teaching role she quickly executed an m_sheet(L) operation to maximise the light intensity which was consistent with an application of Method 1(L). She then inspected the displayed rate/T graph to determine the optimum temperature. The C-row had been set in the maximum position for the L-sheet in Unit-task 1, making an m_row(C) operation unnecessary. However, Sharon did not appreciate that the the C-row was already in the maximum position, and she executed a scope_win operation in preparation for executing an m_row(C) to maximise the C-value.

Unit-task 1

> S: OK, [*pointing with the mouse to the L-value scroll box*] just - the light intensity in carbon dioxide should be at maximum so if we have temperature changing. [*pointing to T-columns on the datasheet*] [*max_win(datasheet) operation executed*] Now with light intensity change it here as well. [*redundant s_display and s_view operations executed*] So if want carbon dioxide at a maximum going to be carbon dioxide maximum, [*m_row(C) operation executed*] light intensity changing.

Unit-task 2

S: Temperature going to change. [*undo max_win(datasheet) operation*] Got graph connected to information right now, so this is what we can change. Also can touch this box [*datacube*] and will change whichever bit of information you need [*pointing to the L-sheet face with the mouse, executing an s_sheet(T) operation (by clicking on the T-sheet face), and executing an s_sheet(L) operation (by clicking on the L-sheet face)*] Who's supposed to be answering these questions, both answering? OK. Bad teacher here. Do you want to use that? If we want going to be changing air temperature to find out maximum. Have line with rows. [*pointing to the datasheet rows(C-values)*]

Unit-task 3

S: OK, light intensity here, [*referring to the L-value scroll box*] clicked so goes to maximum [*m_sheet(L) operation executed*] Not in limited supplies and carbon dioxide has to go to bottom line, so do you want to click on this one here? [*scope-win operation executed*] OK got that line. [*realising that the maximum row position is already selected*]

Unit task 4

S: OK, so, maximum rate would be the peak in this temperature [*referring to currently displayed rate/T graph*] 30 degrees. So that's Number 1.

Episode S3

This episode was concerned with Question 2. It featured applications of Method 3(T) to explore the effect of changing the level of carbon dioxide (Episode S3A) and light intensity (Episode S3B).

Episode S3A

Sharon clearly understood the goal structure and the implementation of Method 3(T). The temperature was set to the optimum temperature in Unit-task 1 by executing m_sheet(T) operations; a technique clearly understood by Sharon, as evidenced by the explanation she gave Uri of this technique. m_row(C) operations were executed in Unit-task 2 without problems.

Unit-task 1

S: OK for this temperature can change if click on temperature [*s_sheet(T) operation executed, followed by an m_sheet(T) operation*] and go to 30 degrees, just gone past one.

U: How can you tell?

S: Here, that's 10 that's 25. [*m_sheet(T) operations executed to demonstrate the change in the T-value associated with the T-sheet (returning to a T-value of 30 degrees) and pointing to the T-value display box in the datasheet window*]

U: OK, I've got you.

Unit-task 2

S: Light changing. If you, carbon dioxide, go down. If you just click on this. [*m_row(C) operation executed; no instance of rate/L graph is displayed as a row (minimum) consisting of zero L-values is highlighted*]

U: Carbon dioxide?

S: Yeah, lets see next line. [*m_row(C) operation executed; no instance of rate/L graph is displayed as a row (following minimum position) consisting of zero L-values is highlighted*]

U: Like that one.

S: If you go down one more. Yeah, each time see it will be increasing each time. [*m_row(C) operations executed and the resulting instances of rate/L inspected*]

Episode S3B

As in Episode S3B Sharon demonstrated an application of Method 3(T) without problems.

Unit-task 1

S: Yeah, each time, see it will be increasing each time. Can do same for light you, you know this display thing here, click on that and put columns, [*s_col(L) operation executed*] change it each time across. [*m_col(L) operation executed*] This button here [*scope_win operation executed*] [*m_col(L) operation followed by the execution of a scope_win and an m_col(L) operation*]. Doesn't seem to be changing very much after 40 degrees.

Episode S4

This episode featured another "expert" application of Method 3(T). Sharon indicated that she was aware of two display configuration techniques when she commented: "You can form different graphs and compare or we can keep changing it on same graph". The second technique (which was adopted without further discussion) is more orientated to a direct manipulation approach. She also demonstrated an understanding of the concept of an optimum value of light intensity when she described it as "going to be where it stops changing, or starts to decrease".

Unit-task 1

S: OK so find out optimum level of light or carbon dioxide. Which one do you want to do?

U: The optimum level of light.

S: OK, so this is light intensity. You can form different graphs and compare or we can keep changing it on same graph. The optimum level is going to be where it stops changing, or starts to decrease.

U: What's this?

S: Optimum level. Got to click back on this. [*scope_win operation executed*] Seems to be increasing quite a bit up to about 40, [*referring to the currently displayed rate/C graph*] do you want to increase it? [*asking Uri if he wants to execute m_col(L) operations to explore the effect of changing the L-value*] Click on [*scope_win operation executed*] Click on 35, [*m_col(L) operation executed*] then up to 40, seems to be increasing and it's gone down. Try 40 again. [*m_col(L) operation executed*] Looks the same doesn't it?

U: 45 is just higher.

S: This is these two figures [*pointing to last two rate values shown in the L-col (intensity value 40)*] 40 degrees temperature, no 40 intensity, said the wrong thing.

Episode S5

This episode featured a highly truncated implementation of Method 6(TT) - the T-sheet was already set at the optimum temperature position, and the variation of the rate with L-value was recalled mentally from Episode S4. Note that Sharon talked about executing operations to explore the effect of increasing the level of carbon dioxide, but as the level was initially set at the maximum, she actually advised Uri to explore the effect of decreasing the C-value.

Unit-task 1

S: Next question. Is there a difference in the effect of increasing levels of light and carbon dioxide on rate of photosynthesis?

U: Only slight difference.

S: Just that it goes up to a point doesn't it? [*referring to the variation of the rate of photosynthesis with light intensity*]

Unit-task 2

S: OK, so if we now increase carbon dioxide levels.

U: Want to increase carbon dioxide?

S: Go to display. [*s_row(C) operation executed*] Rows again. [*m_row(C) operation executed*] Decreases; it just goes on decreasing each time .

U: Yeah

S: So it looks like you can increase carbon dioxide, and the rate will still increase; but with light it comes to a point where it is optimum and stops increasing.

Episode S6

This episode featured a fully truncated application of Method 8(LL) in response to Question 5. The way the rate changed with variations in the light intensity was explored in Unit-task 1 by inspecting the existing rate/L graph, and the variation with level of carbon dioxide was recalled in Unit-task 2 from previous experience (a retrospective application of Method 3(T)). Note that Sharon thought that a line graph was the best in "this case", implying that she preferred a line graph for interpreting animated displays.

Unit-task 1

S: Next question. At the fastest rate of photosynthesis shown on the data, which factor would you try to increase more to attempt to increase the rate of photosynthesis further?

U: Light, I guess.

S: No, it's stopped doing it. [*referring to how the rate of photosynthesis is changing with light intensity*] (mumbles)

Unit-task 2

S: Increase wasn't that big of carbon dioxide, keeps increasing doesn't it.

U: What, when I was increasing this? [*pointing to the C-value rows on the displayed T-sheet. (actually in Unit-task S5/2 Uri was changing the C-value by decreasing it)*]

S: When increase carbon dioxide levels, the levels, the rate also increasing.

U: What, so the more carbon dioxide the faster ...

S: It slows down also. This is really easy. From the information that you have collected - that line [*the displayed rate/L graph*] - you can also look at the data straight off.

U: Oh right, that's all the information. [*Uri pointing to the datasheet window*]

S: This is all the information [*Sharon pointing to the datasheet window*]. You can change type of graph, in this case I think line one's best. Can't really tell with a pie chart can you?

Episode S7

This episode provided an example of the expert use of Method 2(T). Note that Sharon states that "this is a different way of finding it than we did before". She clearly understood that the C-value needed to be at a maximum and that it was necessary to observe the rate-value corresponding to the maximum L-value for the rate/L graph as m_sheet(T) operations were executed.

Unit-task 1

U: Right, um....

S: Its just the temperature changing.

U: So you want to change temperature?

S: Um, starting off, because you've got to make sure that you've got the maximum amount of carbon dioxide and light available.

U: So increase light intensity and ...

S: Carbon dioxide at maximum, so going to be at bottom line carbon dioxide [*Sharon pointing to the maximum C-value row position*] and last bit of light intensity [*referring to maximum L-value shown on the rate/L graph*]. OK, can just click on it. [*Sharon pointing to the maximum C-value row position*]

U: Like that? [*m-row(C) operation executed*]

S: Yeah, now just have to change the temperature, which you can do on the cube top.

U: Go into that cube there.

S: Yeah, just change temperature. [*s_sheet(C) and m_row(T) operations executed in error*] Just press temperature again. Just go back to it. Press temperature. [*s_sheet(T) operation executed*] Start at zero degrees and increase. [*m_sheet(T) operation executed*]

U: OK, is that zero? [*zero T-value for the T-sheet has been selected*]

S: [*Start menu item selected in error*] Click somewhere else. [*Undo start menu item operation executed*]

U: Just take that to nought [*referring to the T-sheet location*] Got to increase ...

S: Increase temperature, see how the rate of carbon dioxide will change [*m_sheet(T) operation executed*].

U: Yes, as you increase temperature, the rate increases.

S: This is light intensity, at maximum carbon dioxide.

U: That's going down. [*referring to the move to the T-sheet position corresponding to the maximum T-value*]

S: It's coming down. This is different way of finding it than we did before [*referring to Episode S2*], but if you go back one, that's the peak

U: Oh, that's maximum.

S: That's maximum.

U: Oh right, 30 degrees is maximum.

S: So that's optimum temperature

U: Is 30 degrees.

Episode S8

This episode consisted of two applications of Method 3(T) to answer Question 2. In Episode S8A the effect of varying the level of carbon dioxide was explored, and in Episode S8B the effect of varying light intensity was explored.

Episode S8A

Uri completed an application of Method 3(T) without any real problems.

Unit-task 1

S: You've got to see way in which varying light and carbon dioxide levels affects the rate. [*m_row(C) operation executed to locate an intermediate row position*]

U: That would be less.

S: Yeah, so if you've got 30 degrees, so if carbon dioxide level, the light intensity levels all varying, how it affects the rate. So if you go back to the top, this button here [*scope_win operation executed to display the row corresponding to the minimum position*] and sort of carry on down. [*m_row(C) operation executed*]

U: What, this is the carbon dioxide? [*referring to the C-values displayed for each row shown in the datasheet window*]

S: Yeah, so you're varying the carbon dioxide levels. Light intensity is just at the bottom, isn't it. [*referring to horizontal axis of the rate/l graph displayed in graph window(1)*] You've got 30 at degrees and the temperature is constant.

U: Yeah

S: Well you're just changing carbon dioxide levels.

U: So go down. [*scope_win operation executed*]

S: So it's increasing.

U: So as you increase carbon dioxide the rate increases as well. Right. [*m_row(C) operation executed*] [*scope_win operation executed*] So this is the peak, yeah? [*Uri pointing to the row corresponding to the maximum C-value*]

S: This is lost. Looks like its increasing. [...]

Episode S8B

Uri instinctively pointed with the mouse to the L-value scroll box to change the L-value. Sharon suggested changing the L-value by changing the column position, implying that she wanted to continue using Method 3(T). He executed a series of interrupted m_col operations; the interruptions being caused by discussion of the effects of the operations on the rate/L graph and a note of the effect of a display bug. Note that Sharon was aware of data omissions.

Unit-task 1

S: You could do the same for changing light intensity by changing the light. Yeah, columns.

U: What by going across to. [*pointing with the mouse to the L-value scroll box in the datacube window*]

S: But just, if you also do it up ...

U: Like this. [*pointing with the mouse to the L-values on the datasheet display*]

S: You know that display, you can just change it to columns. [*s_col(L) operation executed*]

U: So just go across. [*m_col(L) operation executed*]

S: You have to go back a bit. [*scope_win operation executed to display the column corresponding to minimum L-values*]

U: [*m_col(L) operation executed*] What's that? [*Referring to an anomaly in the instance of the rate/C display caused by a null data entry*]

S: It's probably a blank maybe. [*m_col(L) operation executed*] It's also increasing. [*m_col(L) operation executed*]

U: This is light intensity.

S: Your light intensity is changing. [*scope_win operation executed*] I think. It is isn't is it? [*referring to the horizontal axis of the rate/C graph which is labelled as "light intensity" due to a display bug*]

U: [*m_col(L) operation executed*] So if you increase light intensity [*m_col (L) operation executed*]

S: Should be the carbon dioxide level at the bottom. [*Sequence of m_col(L), scope_win and inspect_graph(r/C) operations executed*] It's also increasing, and then sort of stops but really increasing sharply. [*Referring to the shape of the rate/L graph displayed in graph window(1)*] [*m_col(L) operation executed*]

U: Not as sharp.

S: So it's about 35 [*Referring to the L-value at which the gradient of the rate/L graph starts to decrease*]

U: So as you increase light intensity and carbon dioxide level

S: The rate also increasing.

U: Right.

Episode S9

This episode was concerned with Question 3. It consisted of an application of Method 1(T) in Episode S9A to consider the effect of varying the level of carbon dioxide, and an application of Method 3(T) in Episode S9B to explore the effect of varying the light intensity.

Episode S9A

As a rate/C graph corresponding to the optimum temperature was displayed at the start of this episode the application of Method 1(T) was fully truncated, simply consisting of the execution of an inspect_graph(r/C) operation.

Unit-task 1

S: And you've got to find optimum levels of both.

U: And they don't change?

S: You've got to find out what the optimum level is. Where it stops changing, where it stops increasing and starts decreasing or just stops increasing and remains the same rate. If you increase ....

U: Well, stops increasing at about 35. [*referring to the maximum rate shown on the rate/C graph displayed in graph window 1*]

Episode S9B

Sharon asked Uri which variable (C or L) he wanted to observe the effects of changing. He replied "Not bothered". Sharon suggested carbon dioxide, but in fact they explored the effect of changing the light intensity. Note the idiosyncratic way in which Sharon advised Uri to undo his act of minimising graph window(1).

Unit-task 1

S: If you change these [*pointing to the columns corresponding to L-values near the maximum value, and indicating that these L-values need to be increased further to identify the optimum L-value*] on the graph where it stops. [*minimise_win(graph 1)*] That's closed it. I think you are going to have to close this one as well. [*referring to the analysis window*] Just press, try that one. [*Click on graph window(1) icon*] put restore [*undo minimise_win (graph 1)*] Don't worry about other one. [*referring to the analysis window*] Carbon dioxide level changing as well. Try find optimum level of light carbon dioxide, try and find optimum level carbon dioxide, do you want to do that?

U: Not bothered. How do we start?

S: You know the way you were doing it before, you can also try it that way and just keep increasing each time.

U: So go to display.

S: You know that it's going to be at the other end. So do you. [*pointing to the right hand horizontal window scroll box*] [*scope_win operation executed to display the column corresponding to the maximum L-value*].

U: What this? [*referring to the (highlighted) column corresponding to the the maximum L-value*]

S: It goes from 30 onwards, doesn't it? [*m_col(L) and scope_win operations executed*] Keep going, its still 25 [*m_col(L) and scope_win operations executed*] Not really changing that much. Changing a fraction.

Episode S10

This episode was based on recall of previous experience. There is no indication that either Sharon or Uri realised that the maximum value for the light intensity was very close to the non-limiting L-value, while this was not the case for the maximum value of the carbon dioxide level.

Unit-task 1

S: Do you want to go on to next question?

U: Yes.

S: OK, is there any difference in the effect of increasing levels of light and carbon dioxide on the rate of photosynthesis?

U: Yeah there is.

S: Just decreases.

U: Yeah, increases

S: Up to a point, optimum point.

U: Try to find optimum level of either light or carbon dioxide.

S: That's the one we just tried.

U: Question 4.

S: As you said, it just increases as well.

U: If you increase carbon dioxide, the rate increases as well, oh, well........

Episode S11

In response to Uri's answer to Question 5 - "Carbon dioxide I guess" - Sharon used Method 2(C) to demonstrate how the level of carbon dioxide affected the rate of photosynthesis. In Unit-task 1 the temperature was set to the optimum value and in Unit-task 2 the effect of varying the level of carbon dioxide was explored.

Unit-task 1

U: Question five.

S: Yeah

U: Carbon dioxide I guess.

S: Let's go to carbon dioxide. [*s_sheet(C) operation executed followed by m_col(L) and scope_win operations*]

S: Let's suppose the optimum temperature is 30 degrees [*s_row(T) operation executed*] on there [*m_row(T) operation executed*].

Unit-task 2

> S: [...]Keep increasing carbon dioxide levels. [*m_sheet(C) operation executed*] That's increasing as well. [*inspect_graph(r/L) operation executed*]

Ruth/Tom session

In this session all but the first episode featured the use of *Bioview* by Tom with advice from Ruth. In the first session Ruth demonstrated the use of the software. Episodes 2 to 4 were representative of the application of successful methods. Episode 5 was based on recall of previous experience. Episode 6 was a very long and convoluted attempt to answer Question 5, including unsuccessful attempts to implement Method 4 and idiosyncratic methods.

Episode R1

This episode consists of a long unstructured sequence of operations. The top-level goal was to demonstrate the use of the package. Sub-goals, which did not correspond to well defined unit-tasks, were the demonstration of the execution of datacube operations and the use of the datasheet and graph windows. The number of operations executed in error (and hence the length of the interaction string) could be due to the fact that for most of the episode Ruth was trying to demonstrate the operation of the program with respect to a C-row consisting of zero values, resulting in confusion as to why no instances of graphs were obtained.

> R: Clicking on this, got 3 axes, yeah. Just click down, can do the same with all three, bring it up, and to connect data, press connect there. [*pointing to the connect menu item in graph window(1)*] Basically where you have all information.

> T: What does cube represent?

> R: Represents, so if you press on here [*pointing to the Z-sheet scroll box and m-sheet(Z) operation executed*] you can change different amounts of data and you can change axes from putting two over there and moving it about [*s_sheet(X) operation executed followed by s_sheet(Z) operation*] . OK load in file. [*PSYNTH datacube loaded*]. Right [*max_win(datasheet)*] This is about. photosynthesis. You've got light intensity along here, and got temperature which is at zero in corner [*referring to the T-value display box on the datasheet*], down the side you've got levels of carbon dioxide [*referring to the C-rows on the datasheet*].

> T: Right

> R: [*undo max_win(datasheet)*] And you can connect all this information into graphs here. [*con_graph(1) operation executed*]. Right, can change the type

[*s_graph(b) operation executed*] and scale. [*sequence of scale_graph(sheet), scale_graph(row) and {scale_graph(cube)} operations executed*]

T: Can you represent data on graph?

R: Yeah, can also change other temperature by using, [*referring to T-value slider*] or you can use this arrow along here. [*referring to T-value slider, confirming datacube scale, and executing m_sheet(T) operation*]

T: That's at 20 degrees.

R: Yeah, that's going down again. [*referring to the decreasing T-value as m_sheet(T) operations are executed*]

T: Right

R: Change. [*sequence of arbitrary menu selections: {s_graph(b)}, scale_graph(sheet), scale_graph(row)*] Can also have pie charts like that. [*s_graph(p) operation executed followed by a {s_graph(p)} operation*] Can create new graphs like that [*con-graph window(2)*], if you knew that one had ben connected. [*referring to graph window(1)*] you could move it over here. [*referring to the vacant bottom left region of the screen*] You can move things around, like that, [*resize_window(datasheet) operation executed*] so moves over, pressing Tidy tidies them up, [*"click" operation executed to bring the datasheet window to the top of the desk-top*] so can see everything [*Tidy menu item selected*]. Right.

T: Can you actually plot this table here [*referring to the datacube]* on graph there. [*referring to graph window(2)*]?

R: The cube?

T: Yeah

R: Well that's basically, if you [*sequence of s_sheet(L), confirm(scale-cube), and m_sheet(L) operations executed*] can change it around like that, so temperature is now along here [*referring to the T-columns on the datasheet*] [*con_graph(2) operation executed*]

T: Right, so got temperature along the bottom now. [*referring to the horizontal axis of the rate/T graph displayed in graph window(2)*]

R: Yeah can also move it about like this. [*m_row(C) operation executed*] Click light intensity up to make a line graph. [*s_sheet(L) and m_sheet(L) operations executed*]

T: How did you do line graph again?

R: Got type here so just press type. [*{s_graph(l)} operation executed*] If light intensity is at zero nothing on graph, [*m_sheet(L) operation executed*] so either move this along like that, [*m_sheet(L) operation executed - using L-value slider*] or move arrow. [*m_sheet(L) operation executed - using L-value scroll box*].

T: This is with fixed light intensity? At a particular value.

R: Yeah written up here as well. [*referring to the L-value display box in graph window(2)*] At the moment.it's 45. Now 50. [*m_sheet(L) operation executed*] Any questions you'd like to ask?

T: OK, to get maximum point of graph it's not clear to see whether 20, or..

R: If you use bar chart easier to see, [*s_graph(b) operation executed*] yeah.

Episode R2

This episode provided an example of the the application of a truncated version of Method 1(L). The episode started with the L-sheet selected in the position corresponding to the maximum L-value. A row corresponding to an intermediate C-value was highlighted. Ruth correctly and confidently informed Tom that the light intensity was set at a maximum and requested him to select the row corresponding to the maximum C-value. In response to his question as to why this was necessary she explained that the selection of this row gave the maximum value of carbon dioxide. Note that Tom had an incorrect perception of the relationship between the instance of the graph displayed and the datacube representation. He was confusing the C-values represented on the vertical axis of the T-sheet with the rate values represented on the vertical axis of the displayed rate/T graph.

Unit-task 1

R: If we swap seats to answer these questions. Like find the optimum air temperature.

T: Want the maximum.

R: Of carbon dioxide and light, so

T: Want carbon dioxide and light in unlimited supplies.

R: So at the moment ...

T: Here got ...

R: So at the moment got maximum light intensity, which is at 50 and carbon dioxide. We need to, if you go on [*scope_win operation executed*] ... and pull it down, should go down, if keep on going down. Go to 0.1 and click on it.

T: Why?

R: Because if you do that, it gives you the maximum amount of carbon dioxide. So if you click on here, [*m-row(C) operation executed*] it gives you unlimited supplies of light intensity and carbon dioxide.

T: Right, so got carbon dioxide values here, [*referring to the vertical axis of the rate/T graph*] temperature here, [*referring to the horizontal axis of the rate/T graph*] and that's light intensity [*referring to the L-value display in the datasheet window*] which is fixed?

R: Yeah, that's fixed.

T: I think it says make sure neither light nor carbon dioxide are in limited supplies. Would that mean unfixed?

R: It means basically at its maximum. It's not at its minimum.

Unit-task 2

R: If you look at the graph now you can see that 30. [*inspect_graph(r/T) operation executed*]

T: 30, yes, 30 degrees

R: Is the optimum temperature, OK?

Episode R3

This episode was concerned with Question 2. In Episode R3A the effect of varying the level of carbon dioxide was investigated by applying Method 3(T), and in Episode R3B the effect of varying the carbon dioxide was explored by applying Method 3(L). The effect of varying the light intensity was not considered.

Episode R3A

In Unit-task 1 Ruth started to introduce Method 3(L) to explore the effect of changing the the level of carbon dioxide on a rate/T graph by executing m_row(C) operations. However, in Unit-task 2 Tom interrupted to ask how the temperature could be fixed at the optimum value of 30 degrees. Ruth responded by advising Tom to use an m_sheet(T) operation to select the T-sheet corresponding to the optimum temperature. This led to an application of Method 3(T). In Unit-task R3A/3 when the T-sheet was selected the row position defaulted to the minimum position. The row cells in this position all contained zero rate values. This seemed to cause Tom some confusion.

Unit-task 1

T: At this optimum temperature look at the way in which varying ... OK so if keep 30 fixed, go back to....

R: If you want to know levels of carbon dioxide what you could do is click on each of these [*Pointing to the C-rows*] to see what would happen.

Unit-task 2

T: How can I now fix temperature at 50 [*meaning 30 degrees*] so I can look at carbon dioxide and light intensity.

R: What, at 30 degrees, yeah. Need to go back to cube and click on the square for the cube [*referring to the T-value scroll box*]. Need to take that up to 30, you can click on this arrow here. [*s_sheet(T) operation and m_sheet(T) operation executed in succession*]

Unit-task 3

T: I see. So that's on 30.

S: That's on 30, the optimum temperature.

T: Now we can look at graph, can't we?

R: Yeah. [*{con_graph(2)} and {con_graph(2)} operations executed in succession*]

T: Line graph, or bar graph? [*{s_graph(b)} operation executed*] Think bar graph would be better.

R: We need to change [*{restore(graph window(2)} and {s_graph(b)} operations executed in succession*]

T: Scale?

R: No, level of carbon dioxide.

T: Oh right go back to here. [*pointing with the mouse to the C-value scroll box*]

R: Or you could do it from here, [*pointing to the C-rows shown on the datasheet*] just click on again. [*m_row(C operation executed*] Varying light. If vary carbon dioxide levels. [*m_row(C operation executed*]

T: Basically intensity increasing, so could say the light intensity is sort of like proportional to carbon dioxide level.

Episode R3B

Ruth advised Tom to select a L-sheet and execute an m_sheet(L) operation to maximise the light intensity. No actual movement of the L-sheet was required as the L-sheet defaulted to the maximum L-value location. Tom now executed an m_row(C) operation to observe the effect of changing the level of carbon dioxide. This was an application of Method 3(L), which was the method that Ruth suggested when this question was first attempted. Note, however, that the effect of changing the light value was not explored.

Unit-task 1

R: For light intensity back to this square again. [s_sheet*(C) operation executed*].

T: Let's try reducing it. *[m_row(C) operation executed]* Again that's, as you increase carbon dioxide level here, temperature increases and the rate of photosynthesis increases.

R: OK

Episode R4

In Unit-task R4/1 Ruth suggested to Tom that finding the optimum value of light or carbon dioxide was "basically the same as finding same level of temperature". This implied that she thought that Method 1 could be used to determine the optimum value of the light intensity or carbon dioxide level. In Unit-task 1 the level of carbon dioxide was set to a maximum with the temperature at the optimum value. However, in Unit-task 2 the optimum temperature was changed to a maximum value by executing an m_sheet(T) operation. Note that that there is some repetition in Unit-tasks 2 and 3 - both involved an interpretation of the displayed rate/L graph to identify the optimum value of the light intensity. If Method 1(T) had been applied without error the graph would only have been interpreted at the end of Unit-task 3. (Error here applies strictly to the implementation of a chosen method; not to any "cognitive error" associated with a misinterpretation of the task). There was confusion between the idea of a maximum and an optimum value. Tom again misunderstood the relationship between the datacube representation and the graph displays.

Unit task 1

T: Try to find optimum level of either light or carbon dioxide.

R: You can do either one or can do both.

T: So to find this we need to have light or carbon dioxide along this side, *[referring to the vertical axis of the rate/T graph displayed in graph window(2)]* which we've got anyway.

R: No, we'd need light or carbon dioxide along here. *[referring to the horizontal axis of the rate/T graph displayed in graph window(2)]*

T: Along there? Oh I see, right.

R:Along the bottom, so basically the same as finding same level of temperature. So what we need to do is, need to put temperature. If you click on temperature *[s-sheet(T) operation executed]* so got light intensity over here. *[referring to the horizontal axis of the rate/T graph displayed in graph window(2)]*

T: I see, temperature's fixed now at 30. *[referring to the T-value display in graph window(2)]*

R:Yeah.

T: If I try moving along here, shall I? [*m-row(C) operation executed*] Alright, go other way. [*m-row(C) operation executed*]

R: Yeah, need to go back.

T: Over here? [*scroll_win operation executed*]

R: [*m-row(C) operation executed to move the C-row to the maximum position*] Yeah. 0.1.

T: Is that the highest point we can go? [*referring to the maximum C-value for the displayed row*]

R: Yeah, that's maximum there. [*pointing to the rate value for the maximum L-value shown on the horizontal axis of the rate/L graph*]

T: Would seem to be here.

Unit-task 2

R: Yes, haven't got maximum temperature so ...

T: If I go back here. [*referring to the T-value scroll box*]

R: Yeah.

T: Click that [*redundant s_sheet(T) operation executed*] and increase temperature like that. [*m_sheet(T) operation executed*] Right, I think 40 is the maximum. So this one here ... [*referring to the rate value corresponding to the maximum L-value*]

R: Yeah so that is the ...

T: Is there any way we can actually. Hold on. [*sequence of s_graph(l) and s_graph(b) operations executed*] I'm wondering if there is any function on this that allows you to look at maximum and minimum values without having to ... Because I can't see that easily anyway. [*indicating a difficulty in reading the maximum rate value form the displayed graph*] All these here to me look the same.

R: Difficult to tell, but 50 is max.

Unit-task 3

R: Yeah so that is the ... [*indicating that the optimum value of the light intensity is now shown on the rate/L graph*]

Episode R5

This episode did not involve any manipulation of the program. The discussion of Question 4 was limited to brief recall of previous experience.

Unit-task 1

> T: OK. Is there any difference in the effect of increasing levels of light or carbon dioxide to rate of photosynthesis, well there is, we already answered that question from here.
>
> R: Yes, so if you increase levels of both
>
> T: Increases rate of photosynthesis. [...]

Episode R6

Episodes 6A to 6G are all concerned with attempts to answer Question 5. Ruth and Tom experienced considerable confusion, mostly due to a lack of appreciation of the fact that executing an s_sheet operation results in a change the active system register. These episodes provide ample evidence of the significance of the relationship between functional and surrogate models held by the user.

Episode R6A

In this episode the first of three abortive attempts was made to use Method 4. This attempt got no further than an unsuccessful attempt to display two graphs side by side at the same time in order to compare a rate/L graph with a rate/C graph.

At the start of the episode two graphs were open, with graph window (1) completely obscured by graph window (2). It appeared that Ruth thought that only one graph was open as she advised Tom in Unit-task 1 to start a third graph window. This was opened and placed on top of graph window(2). Ruth advised Tom to iconise graph window(3) in order to reveal graph window(2). She then advised Tom to move the location of graph window(2) so as to reveal graph window(1) next to the new position of graph window(2). Note that Ruth used the same idiosyncratic method to do this as Delia in Episode D3.

In Unit-task 2 an appropriate strategy for configuring the screen was discussed. As in Episodes R2 and R4 Tom was confused about the graph axes, thinking that the vertical axes of the graphs represented carbon dioxide and light intensity. Ruth covertly corrected him, suggesting that carbon dioxide should be placed on one horizontal axis and light intensity on the other horizontal axis.

With Ruth's agreement Tom suggested that the temperature should be made the same for each graph. A redundant m_sheet(T) operation was executed in Unit-task 3 in an attempt to achieve this sub-goal. (An attempt was made to maximise the T-sheet when the sheet was already located at the maximum temperature position). This decision indicated a lack of understanding of the relationship between the graphs displayed in graph windows and datacube sheets - providing a graph had at some stage within a unit-task been connected to the active sheet, the graph would correspond to the current sheet value in the sheet sub-register; exercising an m_sheet operation will have no effect in this respect. In Unit-task 3 graph_window(1) had not been connected to the T-sheet during the course of the unit-task, and the graphs displayed in this window (pie chart followed by a bar graph) corresponded to the T-value current in Episode R1.

In Unit-task 4 an attempt was made to change the horizontal axis displayed in graph window(1) to "carbon dioxide". This could have been done by executing an s_col(L) operation. However, Ruth advised executing an s_sheet(C) operation followed by an s_sheet(L) operation. The result of the first operation was to produce a rate/L graph and she commented: "Now that's the same, but we need to change it " Strictly speaking this display was not the same - the rate/L graph displayed in graph window(1) corresponded to row in a T-sheet, and the graph displayed in graph window(2) corresponded to a row in a C-sheet.

Unit task 1

T: Right, at the fastest rate of photosynthesis in data, which factor would you try to increase more, well, can look at that again.

R: You can compare the two.

T: Can we have them on the same sort of, can we split this up into....

R: Yes, if you go up to start, click on graphs, [*start_graph(3) operation executed*] no hang on, go over to down arrow [*minimise_graph window(£) operation executed*], click on it.

T: This gives us the rate of photosynthesis [*pointing to the horizontal axis of the rate/L graph displayed in graph window(2)*] for varying carbon dioxide levels. [*pointing to the vertical axis of the rate/L graph displayed in graph window(2)*]

R: Right if you move that across click on this side here. Move it all the way, [*resize_graph window(2)*] go to the other side and move it all the way across. [*resize_graph window(2)*] Bit fiddly trying to get it.

T: OK shall I put it back there?

R: Yeah, oh, can't see the icons now. Got two graphs here, that's fine. [*graph window(1) is now visible; graph window(2) and graph window(1) are now displayed side by side at the bottom of the screen.*]

T: You can change this one to a bar graph [*s-graph(b) operation executed*] and we can have...

Unit task 2

R: If you want to organise this first, [*referring to graph window(2)*] because that's going to stay frozen. [*referring to the (unconnected) graph window(1)*]

T: We put increasing carbon dioxide here [*referring to the vertical axis of the rate/L graph displayed in graph window(2)*] and light here, so why don't we have it the other way round here? [*referring to graph window(1)*]

R: Yeah, OK.

T: Why don't we put.....

R: What we need is carbon dioxide [*referring to the horizontal axis of the rate/L graph shown in graph window(2)*] along here [*referring to the horizontal axis of the rate/T graph displayed in graph window(1)*]

T: Along the bottom.

R: And then have light intensity along here. [*referring to the horizontal axis of graph window(1)*] No we have light intensity along here [*referring to the horizontal axis of graph window(2)*] and so we need carbon dioxide along here. [*referring to the horizontal axis of graph window(1)*]

Unit task 3

T: Shall we fix temperature to be the same first?

R: Yeah.

T: On both. If I go there. [*redundant m_sheet(T) operation executed]*

Unit task 4

R: No that's not it. [*{start_graph} operation executed*] Press on connect. [*con_graph(1) operation executed*] Now that's the same, [*indicating that the type of graph (rate/L) graph was now the same in both graph windows*] but we need to change it, *referring to graph window(1)*] so if we go over to I think carbon dioxide over here, we click on, [*s_sheet(C) operation executed*] go to light intensity. [*s_sheet(L) operation executed*] right that's temperature [*referring to the rate/T graph now displayed in graph window(1)*], I mean we could.....

Episode R6B

This episode featured the second abortive attempt to apply Method 4. The episode was dominated by Tom's continued misconception of the relationship between the representation provided by the displayed graphs and the datacube. In Unit-task 1 he was concerned with making the horizontal axes of both of the graphs the same. This was attempted in preparation for making one *vertical* axis carbon dioxide and the other light intensity. In Unit-task 2 Ruth corrected his misconception (after first agreeing with his interpretation). She then advised Tom to execute a sequence of operations which left both windows displaying a rate/L graph. This did not match her goal of putting carbon dioxide on one horizontal axis and light intensity on the other horizontal axis. The repetition of redundant con_graph operations indicated the need for action confirmation.

Unit task 1

T: Right, in order for us to compare them I would have thought that this axis here [*referring to the horizontal axis of the rate/l graph displayed in graph window(1)*]would have to be the same to make any sort of...

R: Yeah, I can see what you are getting at.

T: This is a thing that has to vary. [*referring to the vertical axis of both of the displayed graphs*]

R:You mean you can't vary this. [*pointing to the vertical axis of the rate/T graph in graph window(1)*] The only way you can vary this is by changing the ...

T: If we ... Shall we just fix this first? [*referring to the horizontal axis of the rate/L graph displayed in graph window(2)*] Let's either make them both temperature or light intensity, so if you make it, lets connect this one first. [*con-graph(2) operation executed*]

R: They're the same, temperature, everything the same.

Unit task 2

T: Except now we've got ... OK. Now we can vary this one [*referring to the vertical axis of the rate/L graph displayed in graph window(2)*] because these are both ...

R: That's light intensity [*referring to the vertical axis of the rate/L graph displayed in graph window(2)*]

T: Right, see if we can change this one [*referring to the vertical axis of the rate/L graph displayed in graph window(1)*]to carbon dioxide and that ones light intensity [*referring to the vertical axis of the rate/L graph displayed in graph window(2)*] then we can have a look.

R: Yes, I think this is actually the rate of photosynthesis [*referring to the vertical axis of the rate/L graph displayed in graph window(2)*] rather than ... So what you need to do is put light intensity along here, [*referring to the horizontal axis of the rate/L graph displayed in graph window(2)*] carbon dioxide along the bottom there, [*referring to the horizontal axis of the rate/T graph displayed in graph window(1)*] so be able to compare them in rate of photosynthesis. So if we connect this one. [*sequence of three redundant con_graph(2) operations executed*] Right, it's connected. If we put light intensity along here, [*referring to the horizontal axis of the rate/T graph displayed in graph window(1)*] [*two redundant s_sheet(L) operations executed*] we need to go to temperature, go up to temperature. [*s_sheet(T) operation executed*] So that's light intensity.

[*referring to the horizontal axis of the rate/L graph displayed in graph window(2)*] Yeah, if go up to temperature along here and click on arrow up there so we can increase, that's at 40 anyway, that's alright, [*no operation executed*] if you increase level of carbon dioxide, 0.1 anyway [*no operation executed*] if we connect back to this one. [*con_graph(1) operation executed followed by the execution of a redundant con_graph(1) operation*] Right, that's the same now.

T: I thought that this Y axis here, [*referring to the vertical axis of the rate/L graph displayed in graph window(2)*] I thought this [*the datacube*] represented carbon dioxide levels, I thought the whole idea of this was to represent some sort of three-dimensional table, so you could have a look at what happened when you vary three different parameters, so I thought that this here was the carbon dioxide, [*referring to the vertical axis of the rate/L graph displayed in graph window(2)*] that was light intensity and this was your temperature varying. [*referring to the value display bar in window(2)*] So I thought this was just a title. [*the vertical axis title*]

R: Yes, you see you've got a fixed temperature and fixed carbon dioxide level so only thing, that's just showing the rate. [*referring to the vertical axis of the rate/L graph displayed in graph window(2)*]

Episode R6C

A third abortive attempt was made in this episode to apply Method 4. Unit-task 1 was concerned with trying to make the horizontal axis of the graph in graph_window(1) "carbon dioxide". In Unit-task 2 another attempt was made to make the temperatures associated with each graph the same. This diverted the attention of both Ruth and Tom from the use of the sought after screen configuration with two different horizontal graph axes displayed. In Unit-task 1 the rate/C graph displayed in graph window(1) corresponded to a graph of values in the L-sheet sub-register, in which the L-value was a maximum. The rate/C graph displayed in Unit-task 2 corresponded to a graph of values in the T-sheet sub-register, in which the L-value was a minimum. So apparently equivalent graphs were shown, but they corresponded to different sets of variables. Unit-task 2 provided a further example of the need for action confirmation. When the rate/C graph corresponding to the T-sheet was displayed a null graph was produced and Tom instinctively attempted to produce one by executing a redundant s_graph(b) operation. In Unit-task 3 Tom persisted in his misconception about the relationship between the graph display and the datacube display.

Unit-task 1

R: If we connect this graph.

T: Right, if we connect this graph. [*referring to graph window(1)*]

R: Yes.

R: And we change to carbon dioxide on the horizontal axis.

T: OK, can do that by going back here. [*s_sheet(C) operation executed*]

R: Yes, no, try temperature [*s_sheet(T) operation executed*]. I can never remember which way round this goes. No maybe it's light intensity [*s_sheet(L) operation executed*]. No that's temperature. [*referring to the horizontal axis of the rate/T graph displayed in graph window(1)*]

T: Right, want to have carbon dioxide across the bottom?

R: Yes. If we go to display and change it to columns, [*s_col(T) operation executed*] yeah, that's it, so got carbon dioxide along here, [*referring to the horizontal axis of the rate/C graph shown in graph window(1)*] that's basically 0 missed out. [*referring to the initial two zero C-values*]

Unit-task 2

T: I think the temperature is a bit different.

R: That's actually light intensity. [*referring to the value display in graph window(1)*] Need to go to temperature on the arrow. [*referring to the T-value scroll box*]

T: Reduce it?

R: No, increase it to 40.

T: On this one? [*referring to graph window(1)*]

R: Yeah, on that one.

T: Its 50.

R: That's the light intensity.

T: Oh, right.

R: 50 light intensity and 10 is temperature, so if we go back to temperature on arrow up here, [*Tom makes to use the T-value scroll box to decrease the temperature*] on the other arrow [*s_sheet(T) operation executed*] Need to ... yeah that's it, that's at 40.

T: And now we have to ... Shall I do graph, go back to bar graph? [*redundant s_graph(b) operation executed*]

R: And we need to increase light intensity, so if go to light on arrow on bottom, [*s_sheet(L) operation executed*] and take it up to 50 [*m_sheet(L) operation executed; the L-value was initially at 50, Tom decreased the L-value and had to increase it again*] So ...

T: What we've ...

R: So what we've done; go to temperature again on arrow, [*s_sheet(T) operation executed*] and increase that to 40. [*m_sheet(T) operation attempted; T-value already at 40 degrees*] Don't understand why can't have ...

Unit-task 3

T: Shall we go back to what we did with columns?

R: Yes, that's display, go to that.

T: Go back to rows? [*s_row(C) operation executed*]

R: Yes. That's right, got the same up here, [*referring to the C-value and T-value shown in the graph window value displays*] just got two different factors. [*on the horizontal axes*]

T: And this is still ... I mean that is temperature, [*referring to the T-values shown in the graph window value displays*] yeah. So this is carbon dioxide, levels [*referring to the vertical axes of the displayed graphs*], both of these is carbon dioxide.

R: Yeah those two there are rate of photosynthesis, [*referring to the vertical axes of the displayed graphs*] that's carbon dioxide on the bottom, [*referring to the horizontal axis of graph in graph window(1); - this should have been displayed a a rate/L graph, but a display bug showed it as a rate/C graph*] got light intensity along here. [*referring to the horizontal axis of rate/L graph in graph window(2)*]

Researcher: Can I just stop you there, there's a bug in the programme, that really should now read light intensity on the right as well.

Episode R6D

In Unit-task 1 Ruth agreed to Tom's suggestion that the level of carbon dioxide should be fixed and that the temperature should be varied. Estimates were made in Unit-task 2 of the change in the rate (as represented by the magnitude of a chosen bar in the bar graph) as m_sheet(T) operations were executed. This amounted to a decision to adopt Method 7(TT), that is a change in strategy. The original intention in Episode 6A was to compare rate/L and rate/C graphs for the same temperature, but now it was proposed to observe the effect of changing the temperature on a rate/L graph. The problems with manipulating the display had forced a strategy which was not based on an analysis of the task, but simply on the desire to achieve some interpretable results; even if they were not relevant to the current task. In Unit-task 2 Tom suggested that attention should be focussed on the bar in each of the two displayed bar graphs which corresponded to the same L-value. In Unit-task 3 the application of Method 7 was varied, with attention now focussed on one graph only. This was initially graph(1), but was switched to graph(2) so as not to have to cope with the incorrect labelling of the horizontal axis of graph(1) caused by a display bug.

Unit task 1

R: OK. Right. That reads light intensity. [*referring to the horizontal axis displayed in graph window(1)*]

T: Shall we have a look at varying ... Why don't we keep carbon dioxide fixed then, [*pointing to the C-value displayed in the datasheet values display*] and we can vary temperature, that might work?

R: OK

Unit task 2

T: So how can I do that?

R: If you want to vary temperature go to arrow on left that will take it down, [*m_sheet(T) operation executed to decrease the temperature*] that's increasing temperature [*in fact the temperature is decreased*] and light intensity is ...

T: We could just have look at one section of this [*pointing to the datacube*] and say, OK, for something like light intensity 20, [*referring to a specific value of the light intensity on the horizontal axis of the r/L graph displayed in graph window(2)*] so rate of photosynthesis is, what, 20 whatever units you use, [*referring to the corresponding value of the rate on the vertical axis of the r/L graph displayed in graph window(2)*] and now when you the vary light intensity for the same section of the graph, [*that is, the bar in each of the two displayed bar graphs corresponding to the same L-value (in this case 20 units)*] which would be what, it would be this one there wouldn't it, 1,2,3,4; 1, 2, 3, 4. [*identifying the equivalent bar position on each of the two graphs*] It would be this one here, that section there, so if we look at what that section looks like. If you reduce temperature [*m_sheet(T) operation executed to reduce the temperature*] it's getting a lot less, rate of photosynthesis, but by what factor?

Unit task 3

R: By what factor? Um, we are only altering temperature, nothing else, so by decreasing temperature [*m_sheet(T) operation executed to increase temperature*] you are decreasing the rate of photosynthesis. [*m_sheet(T) operation executed to decrease temperature*].

T: The temperature here [*m_sheet(T) operation executed to increase temperature followed by an m_sheet(T) operation to decrease the temperature*] too is increasing by - is that 5 degrees? [*m_sheet(T) operation executed to increase the temperature*]

R: Yeah, 5.

T: So by decreasing it, [*referring to the temperature*] by 5 degrees [*m_sheet(T) operation executed to decrease the temperature by 5 degrees*] it doesn't have that much of an effect.

R: No not that dramatic.

Researcher: Remember you're looking at light intensity there.

R: Yeah. So temperature ...

T: For a fixed light intensity at 20, and just varying temperature; right it's 25, you increase that to 30, [*m_sheet(T) operation executed to change the temperature to 30 degrees*] then if you connect this one it might be easier to look at that. [*con-graph(2) operation executed and attention now focussed on graph window(2)*]

R: So ...

T: [*m_sheet(T) operation executed to decrease the temperature*] It makes ... [*m_sheet(T) operation executed to increase the temperature*]

R: So it just ...

T: By an increase of 10 degrees, [*m_sheet(T) operation executed to change the T-value from 30 degrees to 20 degrees, followed by the execution of m_sheet operations to change the temperature to 10 degrees and back to 20 degrees*] its only varied by well not that much.

R: So I wouldn't say that you would increase temperature.

Episode R6E

At the end of Unit-task 1 a rate/C graph was produced, but the confused nature of the unit-task indicated that Ruth did not appreciate that this graph corresponded to an L-sheet as opposed to a T-sheet. There were a number of arbitrary m_sheet operations executed by Tom, almost random exploratory actions, which were not noticed by Ruth. One m_sheet(C) operation could have been in response to the appearance of a null graph. Unit-task 2 was concerned with a clarification of how "long" numbers were represented on a horizontal axis. Unit-task 3 was a very confused attempt to explore the effect of varying the level of carbon dioxide. As in Unit-task 1 Ruth was confused about the significance of making another sub-register active, and Tom again demonstrated a lack of understanding of the relationship between the datacube representation and the displayed graphs. The display of a null graph again caused problems in Unit-task 3. In Unit-task 4 Ruth advised Tom to maximise the temperature by selecting the T-sheet (change of active sub-register) and executing a m-sheet(T) operation to maximise the T-value. When a L-sheet was selected again (to display a rate/C) graph) Ruth was confused because the temperature was "still" at 10 degrees (the value corresponding to the L-sheet - not the T-sheet).

Unit task 1

T: If you wanted to increase rate of photosynthesis ... Shall we have a look to see what happens if you increase carbon dioxide levels. If you put carbon dioxide across the bottom, [*referring to the horizontal axis of the rate/L graph* *displayed in graph window(2)*] so if you go to, is it this one here? [*referring to the C-value scroll box*]

R: I think so, yeah. If you click.

T: We need to put this carbon dioxide here, [*referring to the left hand C-value edge of the datacube*] don't we. [*referring to the bottom L-value edge of the datacube*] So. [*s_sheet(C) operation executed*]

R: That's light intensity, [*referring to the rate/L graph (for a row in a C-sheet) displayed in graph window(2)*] That's what's confusing. [*s_sheet(L) and m_sheet(L) operations executed*] Go up to temperature. [*s_sheet(T) and m_sheet(T) operations executed*] This always confuses me. What I think ... [*s_sheet(C) operation executed*] What I think, if you go over to the edge of this side here, [*referring to the right hand C-value edge of the datacube*] and go in the middle and click, just click on it, [*s_sheet(T) operation executed*] I can never remember how to do this. It's um ... [*s_sheet(C) operation executed*]. Go to display again and change it, [*m_sheet(C) operation executed*] which is up there, [*pointing to the display menu item in the datasheet window*] back to columns. [*s_col(T) operation executed*] That's temperature. [*referring to the rate/T graph displayed (the currently selected column corresponds to a a set of zero rate values - no graph displayed*] [*m_sheet(C) operation executed*] No, go to light intensity, [*s_sheet(L) operation executed*] that's it.

Unit task 2

T: So we've got a fixed temperature now.

R: Temperature is on 10 degrees at the moment and light intensity is 40.

T: I was just wondering why, how can we get figures along there, the're all zeros aren't they? [*referring to the numbers displayed on the horizontal axis of the rate/C graph in graph window(2); these numbers are shown in a truncated form because of the limited size of graph window(2) in the current screen configuration*]

R: Yeah, should actually be 0.05.

T: If I click this one here, perhaps, how can we click all this. [*referring to the currently highlighted column on the datasheet display*]

R: Can't actually do that. Figures don't come up here. So ...

T: OK, so they should be 0.5 [*meaning 0.05*]

R: Yeah, this should be 0.05 going up to 0.1 which is down here. [*referring to the C-values shown in the datasheet display*]

Unit task 3

T: OK if we increase carbon dioxide, how can we do that. We can increase it from here, can't we? [*referring to the C-value scroll box and s-sheet(C) operation executed*]

R: Yes. See that's by, um ... [*redundant s_graph(b) operation executed: null graph displayed corresponding to a set of zero rate values*] If you increase carbon dioxide again, [*referring to C-value scroll box*] [*m-sheet(C) operation executed*] keep on increasing. [*m_sheet(C) operation executed to decrease the level of carbon dioxide*] It ...

T: It's not plotting. [*m_sheet(C) operation executed*]

R: No [*s_graph(l) operation executed*].No, I don't understand that [*series of redundant operations executed: m_win(graph_window(2) / resize_window(graph_window(2) / output(copy) / con_graph(2) / con_graph(2)*] It's connected. [*redundant s_scale(cube) operation executed*] Go up to display and change it to rows. [*s_row(T) operation executed*] That's it, but we haven't got carbon dioxide at the bottom. [*m_sheet(C) operation executed*] This is one of the problems I found, trying to get the carbon dioxide.

T: If you click ... [*s-sheet(L) operation executed*]

R: If you increase carbon dioxide. If you go over to carbon dioxide and increase that [*referring to the C-value scroll box*]

T: OK, now carbon dioxide is over here, isn't it? [*referring to the vertical axis of the rate/T graph shown in graph window(2)*]

R: No, carbon dioxide is fixed at 0.09. [*referring to the C-value shown in the datasheet value display in graph window(2)*]

I: Right.

R: And this is the rate of photosynthesis up here. [*referring to the rate axis of the rate/C graph shown in graph window(2)*]

T: OK, lets have a look at it this way. If we reduce it. [*referring to the C-value*] [*s_sheet(C) operation executed followed by an m_sheet(C) operation*]

R: Right, at the moment it is at 0.8.

Researcher: If you choose a column now, if it does not change to carbon dioxide it should do. Maybe this is confusing you. [*incorrect advice - a rate/T graph would have been produced by executing an s_col(L) operation; not referred to in Chapter 7*]

T: Right, its decreasing now, [*m_sheet(C) operation executed to decrease the C-value*] decreasing carbon dioxide from 0.07. Go to display choose columns [*s_col(L) operation executed*].

R: Right that's temperature. [*referring to the rate/T graph shown in graph window(2)*]

T: [*s_sheet(L) operation executed*] Carbon dioxide there, [*referring to horizontal axis of the rate/C graph shown in graph window(2)*] Shall we try varying it again? [*referring to the C-value scroll box*

Unit task 4

R: Basically this is ... If we look at this along here. [*referring to the horizontal axis of the rate/C graph displayed in graph window(2)*] If we increase temperature to its maximum. So go to temperature and increase it. [*s_sheet(T) and m_sheet(T) operations executed*] Right go to light intensity and increase that [*s_sheet(L) operation executed*] on the arrow, yes. I don't understand why the temperature is decreasing. [*m_sheet(L) operation executed to maximise the L-value*]

Episode R6F

In Episode R6F Ruth and Tom unsuccessfully applied an idiosyncratic method based on the comparison of a rate/C graph (corresponding to a column selected in a L-sheet) and a rate/T graph (corresponding to a column selected in a C-sheet). The method aimed to make this comparison with the same C- and L-values considered for each graph. An attempt to focus on the same C-value in each graph was made by maximising the C-value (0.1) in graph window(1) by executing an m_sheet(C) command, and focusing attention on the rate value corresponding to the maximum C-value shown on the horizontal axis of the rate/C graph displayed in graph_window(2). The L-value associated with graph window(2) was the L-value in the L-sheet sub-register which was currently at the maximum value. An attempt to focus on the same L-value in each graph was made by maximising the L-value (50) in graph window(1) by executing an m_col(L) operation until the L-column corresponding to the maximum L-value(50) was highlighted. In Unit-task 2, in response to a request from Ruth to increase the level of carbon dioxide, Tom initially executed an m_sheet(C) operation in the direction which decreases the C-value. This provides further evidence of his confusion between the graph and datacube representation.

Unit-task 1

T: Right, that's increasing temperature. Here, 45, 50, maximum ... Just for comparison, can we have, OK, temperature across here and this one here [*referring to the horizontal axis of the rate/L graph displayed in graph window(1)*]

R: Yes, if you connect that graph. [*con_graph(1) operation executed*] Go up to there [*referring to the T-value scroll box*]

T: Right, that's temperature, isn't it?

R: [*s-sheet(T) operation executed*] No that's carbon dioxide. [*s-sheet(L) operation executed*] Press on carbon dioxide. [*s-sheet(C) operation executed*]

T: So now I can display this.

R: Yeah, line graph. [*s_graph(l) operation executed*] [...]

R: We need to increase the light intensity, [*s-sheet(L) operation executed*] so go on arrow again. Right that's the same.

T: But it's gone back to carbon dioxide levels. Go back to [*s_sheet(T) operation executed*]

R: [*s_sheet(L) operation executed*] Go to carbon dioxide. [*s-sheet(C) operation executed*] That's temperature there [*referring to the horizontal axis of the rate/T graph*]

T: Right, so let's try and increase it. [*suggesting that the C-value should be increased*]

R: OK if we move these along, [*referring to the L-columns displayed in the datasheet window*] click on that column [*m_col(L) operation executed*], so

T: Hold on, need to fix this as well. [*referring to the C-value shown in the graph window(1) values display*] This is the carbon dioxide level. Let's have a look at it for 0.1.

R: OK, so go to carbon dioxide and increase it. [*m_sheet(C) operation executed which first decreases the C-value to the minimum value and than increases it to the maximum value*] Right.

T: 0.1 is this value here, isn't it? [*referring to the maximum C-value shown on the horizontal axis of the rate/C graph displayed in graph window(2)*] Last one. So the last one corresponds to about 10 [*referring to the rate value corresponding to the maximum C-value on the horizontal axis of the rate/C graph shown graph window(2)*]. Is this value here the temp? [*pointing to the L-value shown in the datasheet values display in graph window(1)*]

R: No that's light intensity. You could increase light intensity. If you go all the way along here, you click on that. [*m_col(L) operation executed*]

T: So light intensity here is 10? [*referring to T-value shown in the values display for graph window(2)*]

R: No that's the temperature.

T: Yeah.

R: If you go to the arrow there at bottom in the corner [*referring to the right hand horizontal window scroll box*], click on it [*scroll_win operation executed*], keep on clicking, [*scroll_win operation executed*] now click on this column here. [*referring to the L-column corresponding to the maximum L-value*] [*m_col(L) operation executed*] If that's going to be 0.1, [*referring to the maximum L-value shown on the horizontal axis of the rate/L graph displayed in graph window(2)*] light intensity is at 50 [*pointing to both graphs*], and got temperature and carbon dioxide. [*referring to the horizontal axis variable for both graph window(1) and graph window(2) respectively*]

T: This is for 50 as well? [*referring to the L-value shown in the values display for graph window(2)*]

R: Yeah, that's 50. Everything is same. Looking at this, [*referring the portion of the rate/C graph corresponding to the maximum L-value in graph window(2)*] is 0.1. [*referring to the C-value shown in the values display for graph window(2)*] So ...

T: Um ...

3.6.7 Episode R6G

In this episode all the factors (including temperature) were considered. Unit-task 4 featured a discussion of what factor to increase so as to increase the rate the most. This discussion did not involve an evident use of the concept of limiting factors; a numerical estimation method was used instead.

Unit task 1

T: Can I try varying carbon dioxide levels just to get some kind of picture here.

R: Yeah. [*m_sheet(C) operation executed*]

T: I mean, I would ... Because the carbon dioxide levels are increasing by 0.01 each time, which is very small, but even that is having some effect on rate of photosynthesis. [*m_sheet(C) operation executed*]

R: Yes, carbon dioxide seems to ...

Unit task 2

R: If you then connect this graph, [*con_graph(2) operation executed*] that's the same now. [*noting that the same instance of rate/T graph is now shown in both windows*] Oh. [*con_graph(1) operation executed*] Re-connect this graph again, [*con_graph(2) operation executed*] go to light intensity, [*s_sheet(L) operation executed*] right, got carbon dioxide along here, [*referring to the horizontal axis of the rate/C graph displayed in graph window(2)*] um ...

T: So we've got varying temperature now?

R: Yes, so if you vary the temperature. [*s_sheet(T) operation and an m_sheet(T) operation executed in sequence; the T-value is changed to an intermediate value and back to a maximum value*] That's at zero. [*referring to the minimum L-column on the T-sheet, which is currently highlighted*] If you increase it; [*m_sheet(T) operation attempted to increase the temperature*] why is nothing? Oh, its at its maximum, 40. [*m_sheet(T) operation executed to minimise the temperature*]

Unit task 3

R: [...] OK try increasing light intensity. [*s_sheet(L) operation executed*] yeah, or decreasing it [*m_sheet(L) operation executed*] I mean by decreasing light intensity ...

T: When you decrease the light intensity ... I expected that because temperature and this [*referring to the L-value scroll box*] are somewhat related aren't they? Light lets off a certain amount of heat, so if you increase light intensity, it has similar effect to increasing temperatures, [...]

Unit task 4

T: [...] but I don't think either of those two [*that is, light intensity and temperature*] are as good as carbon dioxide.

R: Yeah, carbon dioxide

T: Just look at the factors, [*con_graph(1) operation executed*] increasing carbon dioxide by what, 0.01 each time and yet that is having effect on...

R: So by increasing carbon dioxide rate [*s_sheet(C) operation executed*] that would be the factor to increase.

T: Yes, as it goes up like that, [*m_sheet(C) operation executed to increase the level of carbon dioxide*] still having an effect. [*m_sheet(C) operation executed top decrease the level of carbon dioxide*] So yeah, [*m_sheet(C) operation executed to increase the level of carbon dioxide*] carbon dioxide.

Episode R7

In this episode the design of *Bioview* is discussed by Ruth and Tom.

R: So, that's the programme. Have you got any questions or...?

T: Yeah, this thing here, the cube. It seems like a good idea to use it to represent this sort of model, [*pointing to the datasheet*] but like you said its so difficult.

R: Yes, its confusing.

T: To me it looks like some sort of volumes programme or something, [*pointing to the datacube*], they could have just had three axes or something [*pointing to the datacube edges*] that would have been a lot easier rather than have cube. Could have normal Y and X, and another axis in third dimension [*pointing to the datacube edges*] rather than using this [*that is, the datacube*]. That's just the one thing, apart from that it seems good that you can do all this, can have a look at how two things vary while keeping one thing controlled and so on. I think once you get the hang of it, can be really good.

Appendix 7

Session interaction history records

Delia/Alice session interaction history record

Unit-task	Goal	System register									Display register																
		L-sheet			C-sheet			T-sheet			L-sheet		C-sheet		T-sheet			Graph window (1)					Graph window (2)				
		L	C	T	L	C	T	L	C	T	row	col	row	col	row	col	con	r/L	r/C	r/T	l	b	con	r/L	r/C	r/T	l	b
D1	demo.	2	1	3	2	2	1	1	3	1		•																
D2/1	max.L	3	1	3	2	2	1	1	3	1		•																
D2/2	max C	3	1	3	2	3	1	1	3	1				•														
D2/3	opt. T	3	1	3	3	3	1	1	3	1				•			•			•		•						
D3A/1	open g-win	3	1	3	3	3	1	1	3	1				•			•			•		•		-	-	-	•	
D3A/2	opt. T	3	1	3	3	3	1	1	3	2						•	•		•		•			-	-	-	•	
D3A/3	vary L	3	1	3	3	3	1	3	3	2						•	•		•		•			-	-	-	•	
D3B/1	set T	3	1	3	3	3	1	3	3	2						•	•		•		•			-	-	-	•	
D3B/2	vary C	3	1	3	3	3	1	3	3	2					•			•			•		•	•			•	
D4A/1	opt L	3	1	3	3	3	1	3	3	2					•			•				•	•	•				•
D4B/1	opt. C	3	1	3	3	3	1	3	3	2					•		•	•				•		•				•
D5/1	vary C	3	1	3	3	3	1	3	2	2					•			•				•	•	•			•	
D5/2	vary L	3	1	3	3	3	1	3	2	2						•		•				•	•		•		•	
D6	L/C effect	3	1	3	3	3	1	3	2	2						•		•				•	•		•		•	
D7/1	max. C	3	1	3	3	3	1	3	2	2						•		•				•	•			•	•	
D7/2	max. L	3	1	3	3	3	1	3	2	2		•						•				•	•		•		•	
D7/3	vary T	3	1	2	3	3	1	3	2	2		•					•		•			•			•		•	
D8/1	Max. C	3	1	2	3	3	1	3	3	2					•	•	•	•				•			•		•	
D8/2	vary L	3	1	2	3	3	1	2	3	2						•	•		•			•			•		•	
D8/3	vary T	3	1	2	3	3	1	2	3	2						•	•	•				•			•		•	
D9A/1	locate T	3	1	2	3	2	1	2	3	2						•	•		•			•			•		•	
D9A/2	vary L	3	1	2	3	2	1	3	3	2						•	•		•			•			•		•	
D9B/1	vary C	3	1	2	3	2	1	3	3	2					•		•	•				•			•		•	
D10/1	opt. L	3	1	2	3	2	1	3	3	2						•	•		•			•			•		•	
D11/1	vary L	3	1	2	3	2	1	3	3	2						•	•		•			•			•		•	
D11/2	vary C	3	1	2	3	2	1	3	2	2					•			•					•	•			•	
D11/3	vary L	3	1	2	3	2	1	3	2	2						•	•		•		•			•			•	
D12/1	effect L/C	3	1	2	3	2	1	3	2	2						•	•		•		•			•			•	

Sharon/Uri session interaction history record

Unit-task	Goal	<------------System register------------>									<-----------------------Display register----------------------->																	
		L-sheet			C-sheet			T-sheet			L-sheet		C-sheet		T-sheet		Graph window (1)						Graph window (2)					
		L	C	T	L	C	T	L	C	T	row	col	row	col	row	col	con	r/L	r/C	r/T	l	b	con	r/L	r/C	r/T	l	b
S1	demo.	2	2	1	1	1	2	1	2	1	•						•			•	•							
S2/1	max C	2	3	1	1	1	2	1	2	1	•						•			•	•							
S2/2	illustration	2	3	1	1	1	2	1	2	1	•						•			•	•							
S2/3	max L	3	3	1	1	1	2	1	2	1	•						•			•	•							
S2/4	opt T	3	3	1	1	1	2	1	2	1	•						•		•		•							
S3A/1	opt T	3	3	1	1	1	2	1	2	2					•		•	•			•							
S3A/2	vary C	3	3	1	1	1	2	1	3	2					•		•	•			•							
S3B/1	vary L	3	3	1	1	1	2	3	3	2						•	•		•		•							
S4/1	opt L	3	3	1	1	1	2	2	3	2						•	•		•		•							
S5/1	vary L	3	3	1	1	1	2	2	3	2						•	•		•		•							
S5/2	vary C	3	3	1	1	1	2	2	2	2					•		•	•			•							
S6/1	L effect	3	3	1	1	1	2	2	2	2					•		•	•			•							
S6/2	C effect	3	3	1	1	1	2	2	2	2					•		•	•			•							
S7/1	opt T	3	3	1	1	1	3	3	2	2					•		•	•			•							
S8A/1	vary C	3	3	1	1	1	3	3	2	2					•		•	•			•							
S8B/1	vary L	3	3	1	1	1	3	3	3	2						•	•		•		•							
S9A/1	opt C	3	3	1	1	1	3	3	3	2						•	•		•		•							
S9B/1	opt L	3	3	1	1	1	3	2	3	2						•	•		•		•							
S10/1	L/C effect	3	3	1	2	1	3	2	3	2						•	•		•		•							
S11/1	opt T	3	3	1	2	1	2	2	3	2			•				•			•	•							
S11/2	vary C	3	3	1	2	3	2	2	3	2			•				•			•	•							

Ruth/Tom session interaction history record

Unit-task	Goal	<---System register--->									<---Display register--->																	
		L-sheet			C-sheet			T-sheet			L-sheet		C-sheet		T-sheet		Graph window (1)						Graph window (2)					
		L	C	T	L	C	T	L	C	T	row	col	row	col	row	col	con	r/L	r/C	r/T	l	b	con	r/L	r/C	r/T	l	b
R1	demo	3	2	1	1	1	1	1	1	1	•									•	(p)		•			•		•
R2/1	max C	3	3	1	1	1	1	1	1	1	•									•	(p)		•			•		•
R2/2	opt T	3	3	1	1	1	1	1	1	1	•									•	(p)		•			•		•
R3A/1	vary C	3	3	1	1	1	1	1	1	1	•									•	(p)		•			•		•
R3A/2	optT	3	3	1	1	1	1	1	1	2					•					•	(p)		•	•				•
R3A/3	vary C	3	3	1	1	1	1	1	2	2					•					•	(p)		•	•				•
R3B/1	vary C	3	2	1	1	1	1	1	2	2	•									•	(p)		•			•		•
R4/1	opt L	3	2	1	1	1	1	1	3	2					•					•	(p)		•	•				•
R4/2	max T	3	2	1	1	1	1	1	3	3					•					•	(p)		•	•				•
R4/3	Opt L	3	2	1	1	1	1	1	3	3					•					•	(p)		•	•				•
R5	L/C effect	3	2	1	1	1	1	1	3	3					•					•	(p)		•	•				•
R6A/1	two graphs	3	2	1	1	1	1	1	3	3					•					•		•	•	•				•
R6A/2	spec axes	3	2	1	1	1	1	1	3	3					•					•		•	•	•				•
R6A/3	equalise T	3	2	1	1	1	1	1	3	3					•					•		•	•	•				•
R6A/4	spec axes	3	2	1	1	1	1	1	3	3	•						•			•		•		•				•
R6B/1	spec axes	3	2	1	1	1	1	1	3	3	•									•		•	•			•		•
R6B/2	spec v-axes	3	2	1	1	1	1	1	3	3					•		•	•				•		•				•
R6C/1	spec axes	3	2	1	1	1	1	1	3	3	•						•			•		•		•				•
R6C/2	equalise T	3	2	1	1	1	1	1	3	3						•	•		•			•		•				•
R6C/3	spec axes	3	2	1	1	1	1	1	3	3					•		•	•				•		•				•
R6D/1	fix C	3	2	1	1	1	1	1	3	3					•		•	•				•		•				•
R6D/2	com graphs	3	2	1	1	1	1	1	3	2					•		•	•				•		•				•
R6D/3	est change	3	2	1	1	1	1	1	3	2					•			•				•	•	•				•
R6E/1	rate/C graph	2	2	1	1	3	1	1	3	1		•						•				•	•		•			•
R6E/2	clarify axes	2	2	1	1	3	1	1	3	1		•						•				•	•		•			•
R6E/3	vary C	2	2	1	1	2	1	1	3	1		•						•				•	•		•		•	
R6E/4	insp. graph	3	2	1	1	2	1	1	3	3		•						•				•	•		•		•	

Ruth/Tom session interaction history record (continued)

Unit-task	Goal	<-------------System register								------------>	<------------------------------						Display register								------------------------------>
		L-sheet			C-sheet			T-sheet			L-sheet		C-sheet		T-sheet		Graph window (1)						Graph window (2)					
		L	C	T	L	C	T	L	C	T	row	col	row	col	row	col	con	r/L	r/C	r/T	l	b	con	r/L	r/C	r/T	l	b
R6F/1	rate/T graph	3	2	1	1	2	1	1	3	3				•			•			•	•				•		•	
R6F/2	com graphs	3	2	1	3	3	1	1	3	3				•			•			•	•				•		•	
R6G/1	vary C	3	2	1	3	2	1	1	3	3				•						•	•		•			•	•	
R6G/2	vary T	3	2	1	3	2	1	1	3	3						•				•	•		•		•		•	
R6G/3	vary L	2	2	1	3	2	1	1	3	1		•								•	•		•		•		•	
R6G/4	L/C/T effect	2	2	1	3	2	1	1	3	1				•						•	•		•		•		•	
R7/1	Discusion	2	2	1	3	2	1	1	3	1				•						•	•		•		•		•	

Appendix 8

Task-episode tables

Task-episode table for Question 1

<table>
<tr><th>Episode/
Unit-task</th><th>Goal</th><th colspan="4"><------------------Action string------------------></th><th>Method</th></tr>
<tr><th></th><th></th><th>Observed</th><th>Corrected</th><th>Reduced</th><th>Expert</th><th></th></tr>
<tr><td>D2/1</td><td>max L</td><td>• || s_sheet(T) | s_sheet(L)
| m_sheet(L)</td><td>• || s_sheet(T) | s_sheet(L)
| m_sheet(L)</td><td>• || s_sheet(T) | s_sheet(L)
| m_sheet(L)</td><td></td><td>1(C)</td></tr>
<tr><td>D2/2</td><td>max C</td><td>• || s_sheet(C) | m_sheet(C)</td><td>• || s_sheet(C) | m_sheet(C)</td><td>• || s_sheet(C) | m_sheet (C)</td><td>• || s_sheet(C) | m_sheet(C)
|| s_col(L) | m_col(L)
| inspect_graph(r/T)</td><td></td></tr>
<tr><td>D2/3</td><td>optimum T</td><td>• || s_graph(l) | s_row(T)
| scope_win | s_col(L)
| m_col(L) || con_graph(1)
| inspect_graph(r/T)
| s_graph(b)
| inspect_graph(r/T)</td><td>• || s_graph(l) | s_row(T)
| scope_win | s_col(L)
| m_col(L) || con_graph(1)
| inspect_graph(r/T)
| s_graph(b)
| inspect_graph(r/T)</td><td>• || s_row(T) | s_col(L)
| m_col(L)
| inspect_graph(r/T)</td><td></td><td></td></tr>
<tr><td>D7/1</td><td>max C</td><td>• || s_sheet(C)</td><td>• || s_sheet(C)</td><td>• || s_sheet(C)</td><td></td><td>3(L)</td></tr>
<tr><td>D7/2</td><td>max L</td><td>• || s_sheet(L)</td><td>• || s_sheet(L)</td><td>• || s_sheet(L)</td><td>• || s_sheet(L)</td><td></td></tr>
<tr><td>D7/3</td><td>vary T</td><td>• || con_graph(1) | scope_win
| m_col(T)
| inspect_graph(r/C)</td><td>• || con_graph(1) | scope_win
| m_col(T)
|inspect_graph(r/C)</td><td>• || m_col(T)
|inspect_graph(r/C)</td><td>• || m_col(T)
|inspect_graph(r/C)</td><td></td></tr>
<tr><td>D8/1</td><td>max C</td><td>• || s_sheet(C) | s_sheet(T)
| s_row(C) | m_row(C)</td><td>• || s_sheet(C) | s_sheet(T)
| s_row(C) | m_row(C)</td><td>• || s_sheet(C) | s_sheet(T)
| s_row(C) | m_row(C)</td><td>• || s_sheet(T)</td><td>2(T)</td></tr>
<tr><td>D8/2</td><td>vary L</td><td>• || s_col(L) | scope_win
| m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)</td><td>• || s_col(L) | scope_win
| m_col (L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)</td><td>• || s_col (L) | m_col (L)
| inspect_graph(r/C)</td><td>|| s_col (L) | m_col (L)</td><td></td></tr>
<tr><td>D8/3</td><td>optimum T</td><td>• || m_sheet(T)
| inspect_graph(r/C)</td><td>• || m_sheet(T)
| inspect_graph(r/C)</td><td>• || m_sheet(T)
| inspect_graph(r/C)</td><td>• || m_sheet(T)
| inspect_graph(r/C)</td><td></td></tr>
</table>

Task-episode table for Question 1 (continued)

<table>
<tr><th rowspan="2">Episode/
Unit-task</th><th rowspan="2">Goal</th><th colspan="4"><---------------Action string---------------></th><th rowspan="2">Method</th></tr>
<tr><th>Observed</th><th>Corrected</th><th>Reduced</th><th>Expert</th></tr>
<tr><td>S2/1</td><td>max C</td><td>• || max-win(datasheet)
| {s_display} | {s_view}
| m_row(C)</td><td>• || max-win(datasheet)
| m_row(C)</td><td>• || m_row(C)</td><td>• || m_row(C)</td><td>1(L)</td></tr>
<tr><td>S2/2</td><td>illustration</td><td>• || undo max-win(datasheet)
| s_sheet(T)</td><td>• || undo max-win(datasheet)
| s_sheet(T)</td><td>• || s_sheet(T)</td><td></td><td></td></tr>
<tr><td>S2/3</td><td>max L</td><td>• || s_sheet(L) | m_sheet(L)
| scope_win</td><td>• || s_sheet(L) | m_sheet(L)
| scope_win</td><td>• || s_sheet(L) | m_sheet(L)</td><td>• || s_sheet(L) | m_sheet(L)</td><td></td></tr>
<tr><td>S2/4</td><td>optimum T</td><td>• || inspect-graph(r/T)</td><td>• || inspect-graph(r/T)</td><td>• || inspect-graph(r/T)</td><td>• || inspect-graph(r/T)</td><td></td></tr>
<tr><td>S7/1</td><td>optimum T</td><td>• || inspect_graph(r/L)
| m_row(C) | s_sheet(C)
| m_row(T) | s_sheet(T)
| m_sheet((T) | s_start
| undo s_start | m_sheet(T)
| inspect_graph(r/L)</td><td>• || inspect_graph(r/L)
| m_row(C) | s_sheet(C)
| m_row(T) | s_sheet(T)
| m_sheet((T) | m_sheet(T)
| inspect_graph(r/L)</td><td>• || inspect_graph(r/L)
| m_row(C) | s_sheet(C)
| m_row(T) | s_sheet(T)
| m_sheet((T) | m_sheet(T)
| inspect_graph(r/L)</td><td>• || inspect_graph(r/L)
| m_row(C) | m_sheet(T)
| inspect_graph(r/L)</td><td>2(T)</td></tr>
<tr><td>R2/1</td><td>max C</td><td>• || scope_win | m_row(C)</td><td>• || scope_win | m_row(C)</td><td>• || m_row(C)</td><td>• || m_row(C)</td><td>1(L)</td></tr>
<tr><td>R2/2</td><td>optimum T</td><td>• || inspect_graph(r/T)</td><td>• || inspect_graph(r/T)</td><td>• || inspect_graph(r/T)</td><td>• || inspect_graph(r/T)</td><td></td></tr>
</table>

Task-episode table for Question 2

<table>
<tr><th>Episode/
Unit-task</th><th>Goal</th><th colspan="4"><------------------------------Action string------------------------------></th><th>Method</th></tr>
<tr><th></th><th></th><th>Observed</th><th>Corrected</th><th>Reduced</th><th>Expert</th><th></th></tr>
<tr><td>D3A/1</td><td>open graph
window</td><td>• || start_graph(2) | resize_win
| reseize_win</td><td>• || start_graph(2) | resize_win
| reseize_win</td><td></td><td></td><td>3(T)</td></tr>
<tr><td>D3A/2</td><td>optimum T</td><td>• || s_sheet(T) | m_sheet(T)</td><td>• || s_sheet(T) | m_sheet(T)</td><td>• || s_sheet(T) | m_sheet(T)</td><td>• || s_sheet(T) | m_sheet(T)</td><td></td></tr>
<tr><td>D3A/3</td><td>vary L</td><td>• || {con_graph (1)}
| {con_graph (1)}
| {con_graph (1)}
| {s_col(L)}
| {con_graph (1)}
| m_col(L)
| inspect_graph (r/C)
| s_graph (l)
| m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| scope_win |m_col(L)
| inspect_graph(r/C)</td><td>• || m_col(L)
| inspect_graph(r/C)
| s_graph (l) | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| scope_win |m_col(L)
| inspect_graph(r/C)</td><td>• || s_col(L) | m_col(L)
| inspect_graph(r/C)</td><td>• || s_col(L) | m_col(L)
| inspect_graph(r/C)</td><td></td></tr>
</table>

Task-episode table for Question 2 (continued)

<table>
<tr><th>Episode/
Unit-task</th><th>Goal</th><th><-------------------------------------
Observed</th><th>Action string
Corrected</th><th>
Reduced</th><th>------------------------------------>
Expert</th><th>Method</th></tr>
<tr><td>D3B/1</td><td>optimum T</td><td>• || s_sheet(C) | s_sheet(T)</td><td>• || s_sheet(C) | s_sheet(T)</td><td>• || s_sheet(C) | s_sheet(T)</td><td>• || s_sheet(T)</td><td>3(T)</td></tr>
<tr><td>D3B/2</td><td>vary C</td><td>• || s_col(L) | s_row(C)
| con_graph(2) | m_row(C)
| scope_win | m_row(C)
| m_row(C) | scope_win
| m_row(C)
| inspect_graph(r/L)</td><td>• || s_col(L) | s_row(C)
| con_graph(2) | m_row(C)
| scope_win | m_row(C)
| m_row(C) | scope_win
| m_row(C)
| inspect_graph(r/L)</td><td>• || s_col(L) | s_row(C)
| m_row(C)
| inspect_graph(r/L)</td><td>• || s_row(C) | m_row(C)
| inspect_graph(r/L)</td><td></td></tr>
<tr><td>D9A/1</td><td>optimum T</td><td>• || s_sheet(C) | m_sheet(C)
| s_sheet(T)</td><td>• || s_sheet(C) | m_sheet(C)
| s_sheet(T)</td><td>• || s_sheet(C) | m_sheet(C)
| s_sheet(T)</td><td>• || s_sheet(T)</td><td>3(T)</td></tr>
<tr><td>D9A/2</td><td>vary L</td><td>• || scope_win
| {con_graph(1)} | m_col(L)
| inspect_graph(r/C)
| {con_graph(1)} | m_col(L)
| inspect_graph(r/C)
| {con_graph (1)} | m_col(L)
| inspect_graph(r/C)
| {con_graph(1) | m_col(L)
| scope_win | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph (r/C)</td><td>• || scope_win | m_col(L)
| inspect_graph(r/C)
| m_col(L)
| inspect_graph(r/C)
| m_col(L) |
inspect_graph(r/C)
| m_col(L)
| scope_win | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph (r/C)</td><td>• || m_col(L)
| inspect_graph (r/C)</td><td>• || m_col(L)
| inspect_graph (r/C)</td><td></td></tr>
<tr><td>D9B/1</td><td>vary C</td><td>• || s_row(C) | scope_win
| {m_col(L)} | m_row(C)
| scope_win | m_row(C)
| inspect_graph(r/L)
| scope_win | m_row(C)
| inspect_graph(r/L)</td><td>• || s_row(C) | scope_win
| {m_col(L)} | m_row(C)
| scope_win | m_row(C)
| inspect_graph(r/L)
| scope_win | m_row(C)
| inspect_graph(r/L)</td><td>• || s_row(C) | {m_col(L)}
| m_row(C)
| inspect_graph(r/L)</td><td>• || s_row(C) | m_row(C)
| inspect_graph(r/L)</td><td>3(T)</td></tr>
</table>

Task-episode table for Question 2 (continued)

<table>
<tr><th rowspan="2">Episode/
Unit-task</th><th rowspan="2">Goal</th><th colspan="4"><------------------------------Action string------------------------------></th><th rowspan="2">Method</th></tr>
<tr><th>Observed</th><th>Corrected</th><th>Reduced</th><th>Expert</th></tr>
<tr><td>S3A/1</td><td>optimum T</td><td>• || s_sheet(T) | m_sheet(T)
| m_sheet(T)</td><td>• || s_sheet(T) | m_sheet(T)
| m_sheet(T)</td><td>• || s_sheet(T) | m_sheet(T)</td><td>• || s_sheet(T) | m_sheet(T)</td><td>3(T)</td></tr>
<tr><td>S3A/2</td><td>vary C</td><td>• || m_row(C)
| inspect-graph(r/L)
| m_row(C)
| inspect-graph(r/L)
| m_row(C)
| inspect-graph(r/L)</td><td>• || m_row(C)
| inspect-graph(r/L)
| m_row(C)
| inspect-graph(r/L)
| m_row(C)
| inspect-graph(r/L)</td><td>• || m_row(C)
| inspect_graph(r/L)</td><td>• || m_row(C)
| inspect_graph(r/L)</td><td></td></tr>
<tr><td>S3B/1</td><td>vary L</td><td>• || s_col(L) | m_col(L)
| inspect-graph(r/C)
| scope_win | m_col(L)
| inspect-graph(r/C)
| scope_win | m_col(L)
| inspect-graph(r/C)</td><td>• || s_col(L) | m_col(L)
| inspect-graph(r/C)
| scope_win | m_col(L)
| inspect-graph(r/C)
| scope_win | m_col(L)
| inspect-graph(r/C)</td><td>• || s_col(L) | m_col(L)
| inspect-graph(r/C)</td><td>• || s_col(L) | m_col(L)
| inspect-graph(r/C)</td><td>3(T)</td></tr>
<tr><td>S8A/1</td><td>vary C</td><td>• || m_row(C) | scope_win
| m_row(C)
| inspect_graph(r/L)
| scope_win | m_row(C)
| inspect_graph(r/L)
| scope_win
| inspect_graph(r/L)</td><td>• || m_row(C) | scope_win
| m_row(C)
| inspect_graph(r/L)
| scope_win | m_row(C)
| inspect_graph(r/L)
| scope_win
| inspect_graph(r/L)</td><td>• || m_row(C)
| inspect_graph(r/L)</td><td>• || m_row(C)
| inspect_graph(r/L)</td><td>3(T)</td></tr>
</table>

Task-episode table for Question 2 (continued)

<table>
<tr><th rowspan="2">Episode/
Unit-task</th><th rowspan="2">Goal</th><th colspan="4"><------------------------Action string------------------------></th><th rowspan="2">Method</th></tr>
<tr><th>Observed</th><th>Corrected</th><th>Reduced</th><th>Expert</th></tr>
<tr><td>S8B/1</td><td>vary L</td><td>• || s_col(L) | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| m_col(L)
| inspect_graph(r/C)
| m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)</td><td>• || s_col(L) | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| m_col(L)
| inspect_graph(r/C)
| m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)</td><td>• || s_col(L) | m_col(L)
| inspect_graph(r/C)</td><td>• || s_col(L) | m_col(L)
| inspect_graph(r/C)</td><td>3(T)</td></tr>
<tr><td>R3A/1</td><td>vary C</td><td>• discussion</td><td></td><td></td><td></td><td></td></tr>
<tr><td>R3A/2</td><td>optimum T</td><td>• || s_sheet(T) | m_sheet(T)</td><td>• || s_sheet(T) | m_sheet(T)</td><td>• || s_sheet(T) | m_sheet(T)</td><td>• || s_sheet(T) | m_sheet(T)</td><td>3(T)</td></tr>
<tr><td>R3A/3</td><td>vary C</td><td>• || {con_graph(2)}
| {con_graph(2)}
| {s_graph(b)}
| {restore_win(graph win(2)}
| {s_graph(b)} | m_row(C)
| inspect_graph(r/L)
| m_row(C)</td><td>• || m_row(C)
| inspect_graph(r/L)
| m_row(C)</td><td>• || m_row(C)
| inspect_graph(r/L)</td><td>• || m_row(C)
| inspect_graph(r/L)</td><td></td></tr>
<tr><td>R3B/1</td><td>vary C</td><td>• || s_sheet(L) | m_row(C)
| inspect_graph(r/T)</td><td>• || s_sheet(L) | m_row(C)
| inspect_graph(r/T)</td><td>• || s_sheet(L) | m_row(C)
| inspect_graph(r/T)</td><td>• || s_sheet(L) | m_row(C)
| inspect_graph(r/T)</td><td>3(L)</td></tr>
</table>

Task-episode table for Question 3

<table>
<tr><th rowspan="2">Episode/
Unit-task</th><th rowspan="2">Goal</th><th colspan="4"><------------------------------Action string------------------------------></th><th rowspan="2">Method</th></tr>
<tr><th>Observed</th><th>Corrected</th><th>Reduced</th><th>Expert</th></tr>
<tr><td>D4A/1</td><td>optimum L</td><td>• || s_graph(b)
| inspect_graph(r/L)</td><td>• || s_graph(b)
| inspect_graph(r/L)</td><td>• || inspect_graph(r/L)</td><td>• || inspect_graph(r/L)</td><td>1(T)</td></tr>
<tr><td>D4B/1</td><td>optimum C</td><td>• || s_graph(b) | m_row(C)
| con_graph(1) | m_row(C)
| inspect_graph(r/L)</td><td>• || s_graph(b) | m_row(C)
| con_graph (1) | m_row(C)
| inspect_graph(r/L)</td><td>• || m_row(C)
| inspect_graph(r/L)</td><td>• || m_row(C)
| inspect_graph(r/L)</td><td>3(T)</td></tr>
<tr><td>D10/1</td><td>optimum L</td><td>• || s_col (L) | m_col(L)
| interpret_graph (r/C)
| m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)</td><td>• || s_col (L) | m_col(L)
| interpret_graph (r/C)
| m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)</td><td>• || s_col (L) | m_col(L)
| inspect_graph(r/C)</td><td>• || s_col (L) | m_col(L)
| inspect_graph(r/C)</td><td>3(T)</td></tr>
<tr><td>S4/1</td><td>optimum L</td><td>• || scope_win
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| m_col(L)
| inspect_graph(r/C)</td><td>• || scope_win
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| m_col(L)
| inspect_graph(r/C)</td><td>• || m_col(L)
| inspect_graph(r/C)</td><td>• || m_col(L)
| inspect_graph(r/C)</td><td>3(T)</td></tr>
</table>

Task-episode table for Question 3 (continued)

<table>
<tr><th>Episode/
Unit-task</th><th>Goal</th><th colspan="4"><------------------------------------Action string------------------------------------></th><th>Method</th></tr>
<tr><th></th><th></th><th>Observed</th><th>Corrected</th><th>Reduced</th><th>Expert</th><th></th></tr>
<tr><td>S9A/1</td><td>optimum C</td><td>• || inspect_graph(r/C)</td><td>• || inspect_graph(r/C)</td><td>• || inspect_graph(r/C)</td><td>• || inspect_graph(r/C)</td><td>1(T)</td></tr>
<tr><td>S9B/1</td><td>optimum L</td><td>• || minimise_win(graph 1)
| minimise_win(stats)
|undo minimise_win(graph)
| scope_win
| inspect_graph(r/C)
| m_col(L)
| inspect_graph(r/C)
| scope_win | scope_win
| inspect_graph(r/C)
| m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C</td><td>• || scope_win
| inspect_graph(r/C)
| m_col(L)
| inspect_graph(r/C)
| scope_win | scope_win
| inspect_graph(r/C)
| m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C</td><td>• || m_col(L)
| inspect_graph(r/C)</td><td>• || m_col(L)
| inspect_graph(r/C)</td><td>3(T)</td></tr>
<tr><td>R4/1</td><td>optimum L</td><td>• || s_sheet(T) | m_rowC)
| m_row(C) | scope_win
| m_row(C)
| inspect_graph(r/L)</td><td>• || s_sheet(T) | m_rowC)
| m_row(C) | scope_win
| m_row(C)
| inspect_graph(r/L)</td><td>• || s_sheet(T) | m_rowC)
| inspect_graph(r/L)</td><td>• || s_sheet(T) | m_rowC)
| inspect_graph(r/L)</td><td>1(T)</td></tr>
<tr><td>R4/2</td><td>max T</td><td>• || {s_sheet(T)}
| m_sheet(T)
| inspect_graph(r/L)
| s_graph(l) | s_graph(b)</td><td>• || m_sheet(T)
| inspect_graph(r/L)
| s_graph(l) | s_graph(b)</td><td>• || m_sheet(T)
| inspect_graph(r/L)</td><td></td><td></td></tr>
<tr><td>R4/3</td><td>optimum L</td><td>• || inspect_graph(r/L)</td><td>• || inspect_graph(r/L)</td><td>• || inspect_graph(r/L)</td><td>• || inspect_graph(r/L)</td><td></td></tr>
</table>

Task-episode table for Question 4

<table>
<tr><th>Episode/
Unit-task</th><th>Goal</th><th colspan="4"><----------Action string----------></th><th>Method</th></tr>
<tr><th></th><th></th><th>Observed</th><th>Corrected</th><th>Reduced</th><th>Expert</th><th></th></tr>
<tr><td>D5/1</td><td>vary C</td><td>• || scope_win | m_row(C)
| con_graph (2)
| m_row(C) | s_graph(l)
| m_row(C)
| inspect_graph(r/L)
| scope_win
| m_row(C)
| inspect_graph (r/L)</td><td>• || scope_win | m_row(C)
| con_graph (2)
| m_row(C) | s_graph(l)
| m_row(C)
| inspect_graph(r/L)
| scope_win
| m_row(C)
| inspect_graph (r/L)</td><td>• || m_row(C)
| inspect_graph (r/L)</td><td>• || m_row(C)
| inspect_graph (r/L)</td><td>6(TT)</td></tr>
<tr><td>D5/2</td><td>vary L</td><td>• || s_col(L) | m_col(L)
| scope_win | m_col(L)
| scope_win | m_col(L)
| inspect_graph(r/C)</td><td>• || s_col(L) | m_col(L)
| scope_win | m_col(L)
| scope_win | m_col(L)
| inspect_graph(r/C)</td><td>• || s_col(L) | m_col(L)
| inspect_graph(r/C)</td><td>• || s_col(L) | m_col(L)
| inspect_graph(r/C)</td><td></td></tr>
<tr><td>D11/1</td><td>vary L</td><td>• || scope_win | m_col(L)
| {con_graph(1)}
| m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| scope_win
| inspect_graph(r/C)</td><td>• || scope_win | m_col(L)
| m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| scope_win
| inspect_graph(r/C)</td><td>• || m_col(L)
| inspect_graph(r/C)</td><td>• || m_col(L)
| inspect_graph(r/C)</td><td>6(TT)</td></tr>
<tr><td>D11/2</td><td>vary C</td><td>• || {s_col(L)} | s_row(C)
| scope_win | con_graph(2)
| scope_win | m_row(C)
| inspect_graph(r/L)
| scope_win | m_row(C)
| inspect_graph(r/L)</td><td>• || {s_col(L)} | s_row(C)
| scope_win | con_graph(2)
| scope_win | m_row(C)
| inspect_graph(r/L)
| scope_win | m_row(C)
| inspect_graph(r/L)</td><td>• || {s_col(L)} | s_row(C)
| m_row(C)
| inspect_graph(r/L)</td><td>• || s_row(C) | m_row(C)
| inspect_graph(r/L)</td><td></td></tr>
</table>

Task-episode table for Question 4 (continued)

<table>
<tr><th>Episode/
Unit-task</th><th>Goal</th><th colspan="4"><----------Action string----------></th><th>Method</th></tr>
<tr><th></th><th></th><th>Observed</th><th>Corrected</th><th>Reduced</th><th>Expert</th><th></th></tr>
<tr><td>D11/3</td><td>vary L</td><td>• || m_row(C) | scope_win
| con_graph(1)
| {con_graph(1)}
| s_col(L) | m_col(L)
| s_graph(l) | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)</td><td>• || m_row(C) | scope_win
| con_graph(1)
| s_col(L) | m_col(L)
| s_graph(l) | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)
| scope_win | m_col(L)
| inspect_graph(r/C)</td><td>• || m_row(C) | s_col (L)
| m_col (L)
| inspect_graph(r/C)</td><td>• || s_col (L) | m_col (L)
| inspect_graph(r/C)</td><td>6(TT)
contd.</td></tr>
<tr><td>S5/1</td><td>vary L</td><td>• recall</td><td></td><td></td><td></td><td>6(TT)</td></tr>
<tr><td>S5/2</td><td>vary C</td><td>• || s_row(C) | m_row(C)
| inspect_graph(r/L)</td><td>• || s_row(C) | m_row(C)
| inspect_graph(r/L)</td><td>• || s_row(C) | m_row(C)
| inspect_graph(r/L)</td><td>• || s_row(C) | m_row(C)
| inspect_graph(r/L)</td><td></td></tr>
<tr><td>S10/1</td><td></td><td>• recall</td><td></td><td></td><td></td><td></td></tr>
<tr><td>R5/1</td><td></td><td>• recall</td><td></td><td></td><td></td><td></td></tr>
</table>

Task-episode table for Question 5

<table>
<tr><th>Episode/
Unit-task</th><th>Goal</th><th colspan="4"><——————————Action string——————————></th><th>Method</th></tr>
<tr><th></th><th></th><th>Observed</th><th>Corrected</th><th>Reduced</th><th>Expert</th><th></th></tr>
<tr><td>D6/1</td><td></td><td>• recall</td><td></td><td></td><td></td><td></td></tr>
<tr><td>D12/1</td><td></td><td>• recall</td><td></td><td></td><td></td><td></td></tr>
<tr><td>S6/1</td><td>vary L</td><td>• recall</td><td></td><td></td><td></td><td>8(LL)</td></tr>
<tr><td>S6/2</td><td>vary C</td><td>• || inspect_graph(r/L)</td><td>• || inspect_graph(r/L)</td><td>• || inspect_graph(r/L)</td><td>• inspect_graph(r/L)</td><td></td></tr>
<tr><td>S11/1</td><td>optimum T</td><td>• || s_sheet (C) | m_col(L)
| inspect_graph(r/T)
| scope_win | s_row(T)
| m_row(T)</td><td>• || s_sheet (C) | m_col(L)
| inspect_graph(r/T)
| scope_win | s_row(T)
| m_row(T)</td><td>• || s_sheet (C) | m_col(L)
| inspect_graph(r/T)
| s_row(T) | m_row(T)</td><td>• || s_sheet (C) | s_row(T)
| m_row(T)</td><td>2(C)</td></tr>
<tr><td>S11/2</td><td>vary C</td><td>|| m_sheet (C)
| inspect_graph(r/L)</td><td>• || m_sheet (C)
| inspect_graph(r/L)</td><td>• || m_sheet (C)
| inspect_graph(r/L)</td><td>• || m_sheet (C)
| inspect_graph(r/L)</td><td></td></tr>
<tr><td>R6A/1</td><td>two graphs</td><td>• || start graph(3)
| minimise_win(graph_win(3)
| resize_win(graph win (2)
| resize_win(graph win (2)
| s_graph(b)</td><td>• || start graph(3)
| minimise_win(graph win(3)
| resize_win(graph win (2)
| resize_win(graph win (2)
| s_graph(b)</td><td>-</td><td></td><td>{4}</td></tr>
<tr><td>R6A/2</td><td>specify axes</td><td>• discussion</td><td></td><td></td><td></td><td></td></tr>
<tr><td>R6A/3</td><td>equalise T</td><td>• || m_sheet(T)</td><td>• || m_sheet(T)</td><td>• || m_sheet(T)</td><td></td><td></td></tr>
<tr><td>R6A/4</td><td>specify axes</td><td>• || {start_graph} | con_graph(1) |
| s_sheet(C) | s_sheet(L)</td><td>• || con_graph(1) | s_sheet(C)
| s_sheet(L)</td><td>• || s_sheet(C) | s_sheet(L)</td><td></td><td></td></tr>
<tr><td>R6B/1</td><td>specify axes</td><td>• || con_graph(2)</td><td>• || con_graph(2)</td><td></td><td></td><td></td></tr>
<tr><td>R6B/2</td><td>specify
vertical axes</td><td>• || {con_graph(2)}
| {con_graph(2)}
| {con_graph(2)}
| {s_sheet(L)} | {s_sheet(L)}
| s_sheet(T) | con_graph(1)
| {con_graph(1)}</td><td>• || {s_sheet(L)} | {s_sheet(L)}
| s_sheet(T) | con_graph(1)</td><td>• || s_sheet(T)</td><td></td><td></td></tr>
</table>

Task-episode table for Question 5 (continued)

<table>
<tr><th>Episode/
Unit-task</th><th>Goal</th><th colspan="4"><——————Action string——————></th><th>Method</th></tr>
<tr><th></th><th></th><th>Observed</th><th>Corrected</th><th>Reduced</th><th>Expert</th><th></th></tr>
<tr><td>R6C/1</td><td>specify axis</td><td>• || s_sheet(C)
| inspect_graph(r/L)
| s_sheet(T)
| inspect_graph(r/T)
| s_sheet(L)
| inspect_graph(r/C)
| s_col(T)
| inspect_graph(r/C)</td><td>• || s_sheet(C)
| inspect_graph(r/L)
| s_sheet(T)
| inspect_graph(r/T)
| s_sheet(L)
| inspect_graph(r/C)
| s_col(T)
| inspect_graph(r/C)</td><td>• || s_sheet(C)
| inspect_graph(r/L)
| s_sheet(T)
| inspect_graph(r/T)
| s_sheet(L)
| inspect_graph(r/C)
| s_col(T)
| inspect_graph(r/C)</td><td></td><td>{4}</td></tr>
<tr><td>R6C/2</td><td>equalise T</td><td>• || s_sheet(T)
| inspect_graph(r/C)
| {s_graph(b)}
| s_sheet(L) | m_sheet(L)
| inspect_graph(r/C)
| s_sheet(T)
| inspect_graph(r/C)</td><td>• || s_sheet(T)
| inspect_graph(r/C)
| s_sheet(L) | m_sheet(L)
| inspect_graph(r/C)
| s_sheet(T)
| inspect_graph(r/C)</td><td>• || s_sheet(T)
| inspect_graph(r/C)
| s_sheet(L) | m_sheet(L)
| inspect_graph(r/C)
| s_sheet(T)
| inspect_graph(r/C)</td><td></td><td></td></tr>
<tr><td>R6C/3</td><td>specify axis</td><td>• || s_row(C)
| inspect_graph(r/L)</td><td>• || s_row(C)
| inspect_graph(r/L)</td><td>• || s_row(C)
| inspect_graph(r/L)</td><td></td><td></td></tr>
</table>

Task-episode table for Question 5 (continued)

<table>
<tr><th rowspan="2">Episode/
Unit-task</th><th rowspan="2">Goal</th><th colspan="4"><——————————Action string——————————></th><th rowspan="2">Method</th></tr>
<tr><th>Observed</th><th>Corrected</th><th>Reduced</th><th>Expert</th></tr>
<tr><td>R6D/1</td><td>fix C</td><td>• || inspect_datasheet</td><td>• || inspect_datasheet</td><td>• || inspect_datasheet</td><td>• || inspect_datasheet</td><td>7(TT)</td></tr>
<tr><td>R6D/2</td><td>compare graphs</td><td>• || m_sheet(T) | m_sheet(T)
| compare_graphs(1,2)</td><td>• || m_sheet(T) | m_sheet(T)
| compare_graphs(1,2)</td><td></td><td></td><td></td></tr>
<tr><td>R6D/3</td><td>estimate rate change</td><td>• || m_sheet)T) | m_sheet(T)
| m_sheet(T) | m_sheet(T)
| inspect_graph(r/L)
| m_sheet(T) | m_sheet(T)
| con_graph(2) | m_sheet(T)
| inspect_graph(r/L)
| m_sheet(T)| m_sheet(T)
| inspect_graph(r/L)
| m_sheet(T)
| inspect_graph(r/L)
| m_sheet(T)
| inspect_graph(r/L)</td><td>• || m_sheet)T) | m_sheet(T)
| m_sheet(T) | m_sheet(T)
| inspect_graph(r/L)
| m_sheet(T) | m_sheet(T)
| con_graph(2) | m_sheet(T)
| inspect_graph(r/L)
| m_sheet(T)| m_sheet(T)
| inspect_graph(r/L)
| m_sheet(T)
| inspect_graph(r/L)
| m_sheet(T)
| inspect_graph(r/L)</td><td>• || m_sheet(T)
| inspect_graph(r/L)</td><td>• || m_sheet(T)
| inspect_graph(r/L)</td><td></td></tr>
<tr><td>R6E/1</td><td>Establish rate/C graph</td><td>• || s_sheet(C)
| inspect_graph(r/L)
| s_sheet(L)
| inspect_graph(r/T)
| m_sheet(L) | s_sheet(T)
| inspect_graph(r/L)
| m_sheet(T) | s_sheet(C)
| s_sheet(T)
| inspect_graph(r/L) |
| s_sheet(C) | m_sheet(C)
| m_sheet(C) | s_col(T)
| m_sheet(C) | s_sheet(L)</td><td>• || s_sheet(C)
| inspect_graph(r/L)
| s_sheet(L)
| inspect_graph(r/T)
| m_sheet(L) | s_sheet(T)
| inspect_graph(r/L)
| m_sheet(T) | s_sheet(C)
| s_sheet(T)
| inspect_graph(r/L) |
| s_sheet(C) | m_sheet(C)
| m_sheet(C) | s_col(T)
| m_sheet(C) | s_sheet(L)</td><td>• || s_sheet(C)
| inspect_graph(r/L)
| s_sheet(L)
| inspect_graph(r/T)
| m_sheet(L) | s_sheet(T)
| inspect_graph(r/L)
| m_sheet(T) | s_sheet(C)
| s_sheet(T)
| inspect_graph(r/L) |
| s_sheet(C) | m_sheet(C)
| m_sheet(C) | s_col(T)
| m_sheet(C) | s_sheet(L)</td><td></td><td>{Idio}</td></tr>
</table>

Task-episode table for Question 5 (continued)

<table>
<tr><th>Episode/
Unit-task</th><th>Goal</th><th colspan="4"><--Action string--></th><th>Method</th></tr>
<tr><th></th><th></th><th>Observed</th><th>Corrected</th><th>Reduced</th><th>Expert</th><th></th></tr>
<tr><td>R6E/2</td><td>Clarify axes</td><td>• || inspect_graph window(2)
| inspect_datasheet</td><td>• || inspect_graph window(2)
| inspect_datasheet</td><td>• || inspect_graph window(2)
| inspect_datasheet</td><td></td><td>{Idio. contd.}</td></tr>
<tr><td>R6E/3</td><td>Vary C</td><td>• || s_sheet(C)
| inspect_graph(r/T)
| {s_graph(b)} | m_sheet(C)
| m_sheet(C) | m_sheet(C)
| inspect_graph(r/T)
| s_graph(l)
| m_win(graph window(2))
| resize_win(graphwindow(2))
| {output(copy)
| {con_graph(2)}
| {con_graph(2))}
| {scale(cube)} | s_row(T)
| inspect_graph(r/L)
| m_sheet(C)
| inspect_graph(r/L)
| s_sheet(L) | s_sheet(C)
| m_sheet(C) | m_sheet(C)
| inspect_graph(r/L)
| s_col(L)
| inspect_graph(r/T)
| s_sheet(L)
| inspect_graph(r/C)</td><td>• || s_sheet(C)
| inspect_graph(r/T)
| m_sheet(C) | m_sheet(C)
| m_sheet(C)
| inspect_graph(r/T)
| s_graph(l)
| m_win(graph window(2))
| resize_win(graphwindow(2))
| s_row(T)
| inspect_graph(r/L)
| m_sheet(C)
| inspect_graph(r/L)
| s_sheet(L) | s_sheet(C)
| m_sheet(C) | m_sheet(C)
| inspect_graph(r/L)
| s_col(L)
| inspect_graph(r/T)
| s_sheet(L)
| inspect_graph(r/C)</td><td>• || s_sheet(C)
| inspect_graph(r/T)
| m_sheet(C) | m_sheet(C)
| m_sheet(C)
| inspect_graph(r/T)
| s_graph(l)
| m_win(graph window(2))
| resize_win(graphwindow(2))
| s_row(T)
| inspect_graph(r/L)
| m_sheet(C)
| inspect_graph(r/L)
| s_sheet(L) | s_sheet(C)
| m_sheet(C) | m_sheet(C)
| inspect_graph(r/L)
| s_col(L)
| inspect_graph(r/T)
| s_sheet(L)
| inspect_graph(r/C)</td><td></td><td></td></tr>
<tr><td>R6E/4</td><td>Interpret rate/C graph at max T</td><td>• || s_sheet(T) | m_sheet(T)
| s_sheet(L)
| inspect_graph(r/C)
| m_sheet(L)</td><td>• || s_sheet(T) | m_sheet(T)
| s_sheet(L)
| inspect_graph(r/C)
| m_sheet(L)</td><td>• || s_sheet(T) | m_sheet(T)
| s_sheet(L)
| inspect_graph(r/C)
| m_sheet(L)</td><td></td><td></td></tr>
</table>

Task-episode table for Question 5 (continued)

<table>
<tr><th>Episode/
Unit-task</th><th>Goal</th><th colspan="4"><--------------------Action string--------------------></th><th>Method</th></tr>
<tr><th></th><th></th><th>Observed</th><th>Corrected</th><th>Reduced</th><th>Expert</th><th></th></tr>
<tr><td>R6F/1</td><td>Display rate/T graph</td><td>• || con_graph(1) | s_sheet(T)
| s_sheet(L) | s_sheet(C)
| s_graph(l)</td><td>• || con_graph(1) | s_sheet(T)
| s_sheet(L) | s_sheet(C)
| s_graph(l)</td><td>• || con_graph(1) | s_sheet(T)
| s_sheet(L) | s_sheet(C)
| s_graph(l)</td><td></td><td>{Idio}</td></tr>
<tr><td>R6F/2</td><td>Compare graphs</td><td>• || s_sheet(L)
| inspect_graph(r/C)
| s_sheet(T)
| s_sheet(L) | s_sheet(C)
| inspect_graph(r/C)
| m_col(L) | m_sheet(C)
| compare_graphs(1,2)
| m_col(L) | scroll_win
| scroll_win | m_col(L)
| compare_graphs(1,2)</td><td>• || s_sheet(L)
| inspect_graph(r/C)
| s_sheet(T)
| s_sheet(L) | s_sheet(C)
| inspect_graph(r/C)
| m_col(L) | m_sheet(C)
| compare_graphs(1,2)
| m_col(L)
| compare_graphs(1,2)</td><td>• || s_sheet(L)
| inspect_graph(r/C)
| s_sheet(T)
| s_sheet(L) | s_sheet(C)
| inspect_graph(r/C)
| m_col(L) | m_sheet(C)
| compare_graphs(1,2)
| m_col(L)
| compare_graphs(1,2)</td><td></td><td></td></tr>
<tr><td>R6G/1</td><td>vary C</td><td>• || m_sheet(C) | m_sheet(C)
| con_graph(2) | con_graph(1)
| con_graph(2) | s_sheet(L)</td><td>• || m_sheet(C) | m_sheet(C)
| con_graph(2) | con_graph(1)
| con_graph(2) | s_sheet(L)</td><td>• || m_sheet(C) | m_sheet(C)
| con_graph(2) | con_graph(1)
| con_graph(2) | s_sheet(L)</td><td></td><td>{Idio.}</td></tr>
<tr><td>R6G/2</td><td>vary T</td><td>• | s_sheet(T) | m_sheet(T)
| m_sheet(T)</td><td>• || s_sheet(T) | m_sheet(T)
| m_sheet(T)</td><td>• || s_sheet(T) | m_sheet(T)
| m_sheet(T)</td><td></td><td></td></tr>
<tr><td>R6G/3</td><td>vary L</td><td>• || s_sheet(L) | m_sheet(L)</td><td>• || s_sheet(L) | m_sheet(L)</td><td>• || s_sheet(L) | m_sheet(L)</td><td></td><td></td></tr>
<tr><td>R6G/4</td><td>Identify factor</td><td>• || con_graph(1) | s_sheet(C)
| m_sheet(C) | m_sheet(C)
| m_sheet(C)</td><td>• || con_graph(1) | s_sheet(C)
| m_sheet(C) | m_sheet(C)
| m_sheet(C)</td><td>• || con_graph(1) | s_sheet(C)
| m_sheet(C) | m_sheet(C)
| m_sheet(C)</td><td></td><td></td></tr>
</table>